

# Accounts Receivable System

Catalog Number 26-1555



**SOFTWARE**

---

The Radio Shack Accounts Receivable System will produce professional Receivables statements, aging reports and automatic General Ledger posting. Each data diskette can handle up to 300 accounts with 1000 transactions per month. An option provides for 100 accounts with 2000 transactions per month\*. The ARS is designed for use with the Radio Shack "32K Business System":

- TRS-80 16K Level II Keyboard
- TRS-80 16K Expansion Interface
- Two TRS-80 Disk Drives\*
- TRS-80 Tractor-Feed Printer
- TRS-80 Video Monitor

\*Three-Disk Systems will raise the capacity to 500 Accounts and 2500 Transactions.


**The Accounts Receivable System Features:**

- Complete End-Of-Month Billing
- Billing Statements Ready For Mailing
- Automatic Customer Records Updating
- Total For General Ledger Postings
- Optional Advertising Message On Billing Statements
- Full A/R Analysis Including Aging Status



# **Accounts Receivable System**

**Radio Shack<sup>®</sup>**

 A DIVISION OF TANDY CORPORATION  
FORT WORTH, TEXAS 76102

**First Edition – 1979**

All rights reserved. Reproduction or use, without express permission, of editorial or pictorial content, in any manner, is prohibited. No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this book, the publisher assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

© Copyright 1979, Radio Shack  
A Division of Tandy Corporation,  
Fort Worth, Texas 76102, U.S.A.

# Table Of Contents

<b>Introduction .....</b>	<b>1</b>
<b>How To Use The ARS .....</b>	<b>1</b>
<b>Features Of The ARS .....</b>	<b>1</b>
<b>Some Decisions You Must Make .....</b>	<b>3</b>
<b>Storage Capacity .....</b>	<b>3</b>
<b>General Ledger Code Numbers .....</b>	<b>3</b>
<b>Setting Up ARS .....</b>	<b>5</b>
<b>How Many Customers And Transactions .....</b>	<b>9</b>
<b>Password Protection .....</b>	<b>9</b>
<b>Running ARS .....</b>	<b>11</b>
<b>Add Accounts .....</b>	<b>12</b>
<b>New Account .....</b>	<b>13</b>
<b>Old Account .....</b>	<b>15</b>
<b>Customer Accounts .....</b>	<b>17</b>
<b>Search .....</b>	<b>17</b>
<b>Previous .....</b>	<b>18</b>
<b>Next .....</b>	<b>18</b>
<b>Turn Page .....</b>	<b>18</b>
<b>Update .....</b>	<b>18</b>
<b>Hard Copy .....</b>	<b>19</b>
<b>Delete .....</b>	<b>19</b>
<b>Transactions .....</b>	<b>20</b>
<b>Sales .....</b>	<b>20</b>
<b>Payment .....</b>	<b>21</b>
<b>Bad Debt .....</b>	<b>21</b>
<b>How To Make Correction Entries .....</b>	<b>22</b>
<b>Crediting A Customer's Account For Returns .....</b>	<b>22</b>
<b>Post Transactions .....</b>	<b>23</b>
<b>List All Transactions .....</b>	<b>24</b>
<b>Post All Transactions .....</b>	<b>24</b>
<b>General Ledger Recap .....</b>	<b>24</b>
<b>Reports .....</b>	<b>25</b>
<b>Entire Account Records .....</b>	<b>25</b>
<b>Status Aging Categories .....</b>	<b>26</b>
<b>Aging Status Analysis .....</b>	<b>27</b>
<b>Current Balance Records (Over Credit Limit) .....</b>	<b>27</b>

<b>End Of Period Processing .....</b>	<b>28</b>
<b>Billing Statement Message .....</b>	<b>28</b>
<b>Period Ending Date .....</b>	<b>28</b>
<b>Align Printer For Statements .....</b>	<b>30</b>
<b>Print All Statements .....</b>	<b>31</b>
<b>Update .....</b>	<b>32</b>
<b>Single Statement Printout .....</b>	<b>32</b>
<b>System Status .....</b>	<b>33</b>
<b>Exit Program .....</b>	<b>33</b>
<b>The @ Key (In General) .....</b>	<b>33</b>
<b>The ← Key .....</b>	<b>33</b>
<b>Appendices</b>	
<b>A – How To Format Your Data Diskettes .....</b>	<b>35</b>
<b>B – How To Backup Your Diskettes .....</b>	<b>36</b>
<b>C – Rotating Your Copies .....</b>	<b>37</b>
<b>D – Statement Alignment Mask .....</b>	<b>38</b>
<b>E – Sample Reports .....</b>	<b>39</b>
<b>Statement Order Form</b>	
<b>Program Listing</b>	

## Introduction

If you consider the General Ledger as the heart of the accounting department, then Accounts Receivable must be considered the pulse of the company. The flow of cash (from goods sold) must be monitored precisely to keep a company healthy. Anything that will assist in getting statements out to customers quickly, and accurately, can be a vital aid to a company. The Radio Shack Accounts Receivable System (ARS) is just such an aid.

The ARS will produce professional-looking statements, aging reports, and automatic totals for posting to your company's General Ledger. The ARS will tell you exactly how long your customers take to pay their bills.

## How To Use The ARS

After you "setup" the Accounts Receivable System, daily operations will consist of: entering transactions, posting transactions, and adding new customer accounts.

Once a month, the "End-of-Period Processing" will take place. This function prints customer billing statements. It will update each account's status automatically. (Reports and summaries can be printed at any time.)

## Features Of The ARS

1. Complete Billing Operation At The End Of The Month.
2. Prints Billing Statements For Mailing.
3. Automatic Updating Of Customer Records.
4. Prints The Following Reports:
  - Complete Transaction File Report
  - General Ledger Recap Report
  - Complete Account Listing
  - Account Listing By Aging Status
  - Accounts Receivable Analysis By Aging Status
  - Posting Report
5. Provides Totals For General Ledger Postings.
6. Advertising Message Option On Billing Statements.
7. Easy Access To Customer Accounts For Updating, Deleting, or Printing.



### Storage Capacity

Before you can enter data, you will be asked to choose which of three Storage Capacity Options is best suited to your needs. Once you choose the storage capacity option, you cannot change the capacity until "End of Period Processing". At that point, you can increase the number of accounts if you desire. You cannot decrease the account capacity.

Two Disk System Options:

1. 300 accounts and 1000 transactions per month.
2. 100 accounts and 2000 transactions per month.

Three Disk System Option:

3. 500 accounts and 2500 transactions per month.

### General Ledger Code Numbers

Any Accounts Receivable System must have some method of keeping track of transactions (for posting to a company's General Ledger). The Radio Shack ARS uses a General Ledger "Code Number". For each GL "Code Number" (1-24), there must be a corresponding General Ledger "Account Number". Your firm's General Ledger account number (up to four digits) will be used by ARS to generate totals for each General Ledger account.

A five letter abbreviation of the General Ledger account name can be included in the GL "code". For example: "14—CASH" could be the GL "code" for a General Ledger "CASH SALES" account.

If your company has General Ledger account numbers larger than four digits, you might want to use the four most significant numbers. For example: you could change 23001 to 2301 or 3001.

Consider using Radio Shack's General Ledger 1.1 in conjunction with ARS since it is designed for four digit GL account numbers. Radio Shack's ARS adheres to generally accepted accounting principles.



## Setting Up ARS

Before you begin using the Accounts Receivable System, you should make a copy (BACKUP) of your Program Diskette. Keep your original Program Diskette in a safe place. You must also prepare (FORMAT) a Data Diskette (which will be used to store your customer accounts and each month's transactions).

If you are not familiar with BACKUP and FORMAT functions, refer to Appendices A and B (at the back of this manual) for instructions.

The ARS "SETUP" program is used only once to enter information about your company into the computer. After the "Set Up" is complete, you will run "ARS" every time you use the system. (If you need to modify your company data, the ARS "SETUP" program has a built-in "Modification" option.)

Follow these steps in setting up the Accounts Receivable System:

1. Turn on the Disk Drives, Printer, Video Display and the Expansion Interface.
2. Insert the ARS Backup Program Diskette in Drive #0 and close the door.
3. Insert the formatted Data Diskette in Drive #1 and close the door. (If you have a three-Drive system, place another formatted Data Diskette in Drive #2.)
4. Turn on the TRS-80 Keyboard.
5. The screen will show:

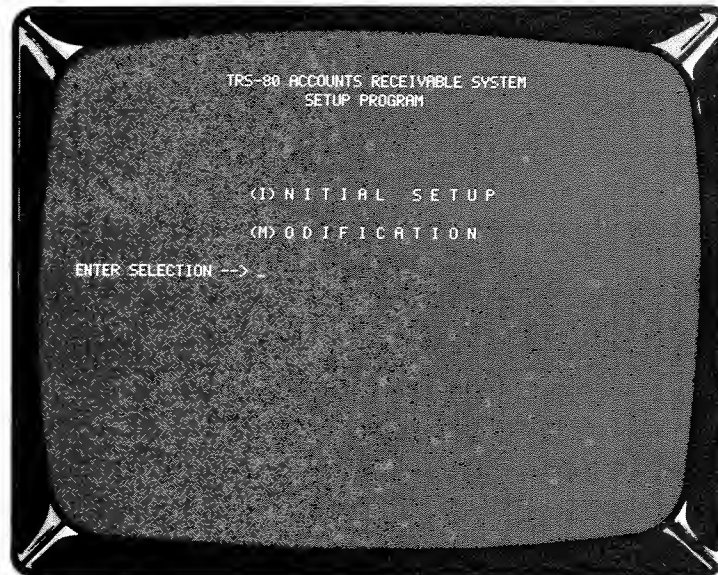
DOS READY  
HOW MANY FILES?\_  
MEMORY SIZE?\_  
READY >\_

This is what you type:

**BASIC** and press **ENTER**  
Press **ENTER**  
\*Press **ENTER**  
**RUN "SETUP"** and press **ENTER**

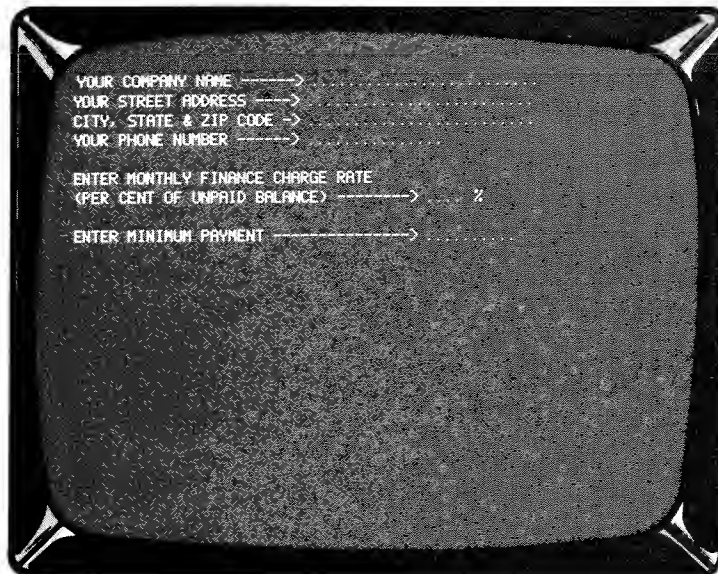
## Set Up (continued)

The screen will show:



This is the Initial Set Up, so press **I**.

The screen will show:



You can now tell ARS a little about your company and its policies concerning Accounts Receivable. Type the information as it is requested by the program. The small blocks indicate the amount of space available for each entry.

Press **ENTER**, after you finish each line. Most entries will require you to press **ENTER** after you type in the information. If you are just answering a Y/N question, or selecting functions, you simply press the appropriate key — the TRS-80 will do the rest.

Do not type a percent sign for the monthly finance charge rate. When you enter the finance charge rate, the computer will automatically add a % sign to the number. For example, if you type 1.5, then after 30 days, the customer will be charged 1½% of the unpaid balance each month as a late charge.

The “Minimum Payment” is an amount your firm uses as a “cut-off” point. Any balance owed below the “minimum payment” must be paid in full. An amount above the “minimum payment” may be paid in pre-arranged installments. These installments are a percentage of the total balance. The payments can be different for every customer. If the installment payment works out lower than the “minimum payment”, the customer will be charged the “minimum payment” each month. See page 14.

After Minimum Payment, the program will ask:

ARE YOU USING BILLING STATEMENT FORMS WITH YOUR  
COMPANY NAME PRE-PRINTED ON THE FORM? (Y/N) \_

Just press the **Y** key for Yes, or the **N** key for No.

The screen will show:

ASSIGNMENT OF GENERAL LEDGER CODES (GLC)  
GLC: 1 & 2 ACCOUNTS RECEIVABLE (A/R) GL ACCOUNT #:

GLC 1 and 2 are the General Ledger “Code Numbers” (GLC’s) for “Accounts Receivable”. Code #1 is for Debits, Code #2 for Credits. You don’t have to concern yourself with the Debit/Credit part — ARS will keep it straight. Just enter your General Ledger “Accounts Receivable” account number. (These two codes are used internally by ARS.)

The computer will ask you to enter General Ledger account numbers for “Assessed Finance Charges” (General Ledger Code #3) and “Bad Debt Write-Off” (General Ledger Code #4).

## Set Up (continued)

The next 20 GLC numbers are optional. You may assign each of these "codes" a General Ledger account number, and a five letter abbreviated account description.

Your firm should already have General Ledger accounts for Cash, Sales, Sales Discounts, Sales Taxes, Shipping Charges, and various other categories. All you do is enter the General Ledger account number, and an abbreviated account description.

The computer will match the General Ledger account number (and account description) with the assigned General Ledger Code. When entering transactions, type in a General Ledger Code number. The TRS-80 will then display the General Ledger Code number, the account description, and a request for the amount. For example:

GLC 5    SLTAX    AMOUNT: \_\_\_\_\_

By using General Ledger Codes, the activity for a General Ledger account is accumulated, and current totals can be printed. At the end of the month, all totals can easily be posted to the proper General Ledger accounts.

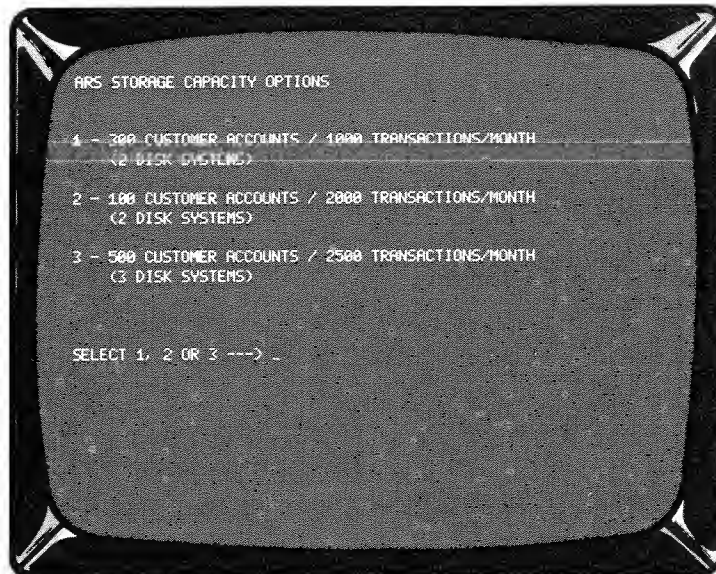
Why didn't we use General Ledger account numbers as "codes" for processing transactions? The one- or two-digit code numbers are easier (and quicker) to type than account numbers with four digits. When you've finished entering General Ledger Codes, type ☐ 1 to end this section.

At this point, you can printout a record of your General Ledger Codes, General Ledger Account Numbers, and General Ledger Account Descriptions. You'll need them for reference when you enter transactions.

## How Many Customers And Transactions

You can now select the "ARS Storage Capacity" option you wish to use.

The screen will show:



Remember, this option can only be changed after "End of Period Processing". If you do decide to change, you can increase the number of accounts — you cannot decrease. Type the number of the option you need.

## Password Protection

The screen will show:

```
ENTER YOUR PASSWORD --->
```

Use of a password will insure that only authorized personnel can gain access to the Accounts Receivable records. If your company does not require a secured ARS, type **PASSWORD** as your "password". (It's easy to remember.)

Your password can be up to eight characters. It can be a combination of letters and numbers, but the first character must be a letter. Press **ENTER** after typing the last letter.

**Note:** This is the only time the password will appear on the screen exactly as typed. From this point on, the password will appear as a series of number signs (#####) whenever you enter your password.

**WARNING:**

REMEMBER YOUR PASSWORD. IF YOU FORGET THE PASSWORD, YOU MAY NEVER SEE YOUR ACCOUNTS RECEIVABLE RECORDS AGAIN.

The TRS-80 will ask:

ARE ALL ENTRIES CORRECT? (Y/N)

If you press **N** for No, the program will display everything you've entered and let you change the Set Up data. If you press **Y** for Yes, the system will store all of the company's data, together with your password.

You are now ready to use the ARS program. If you are ready to start entering your customer accounts, go to Page 12.

If you plan to enter the customer accounts later, remove the Diskettes and place them in their protective sleeves. Shut off the computer. When you are ready to use ARS, you will start on Page 11.

## Running ARS

Follow these steps in exact order:

1. Turn on the Disk Drives, Printer, Video Display and the Expansion Interface.
2. Insert the ARS Backup Program Diskette in Drive #0 and close the door.
3. Insert the formatted Data Diskette in Drive #1 and close the door. If you have a three-Drive system, insert another formatted Data Diskette in Drive #2 and close the door.
4. Turn on the TRS-80 keyboard.
5. 

<b>The screen will show:</b>	<b>This is what you type:</b>
DOS READY	<b>BASIC</b> and press <b>ENTER</b>
HOW MANY FILES?_	Press <b>ENTER</b>
MEMORY SIZE?_	*Press <b>ENTER</b>

The screen will show:

READY >\_

Type: **RUN** **ARS** and press **ENTER**.

The screen will show:

ENTER CORRECT DATE (MM/DD/YY):

Enter today's date (Example: 05/07/79). The date will appear on the screen with the question:

IS THIS CORRECT? (Y/N)

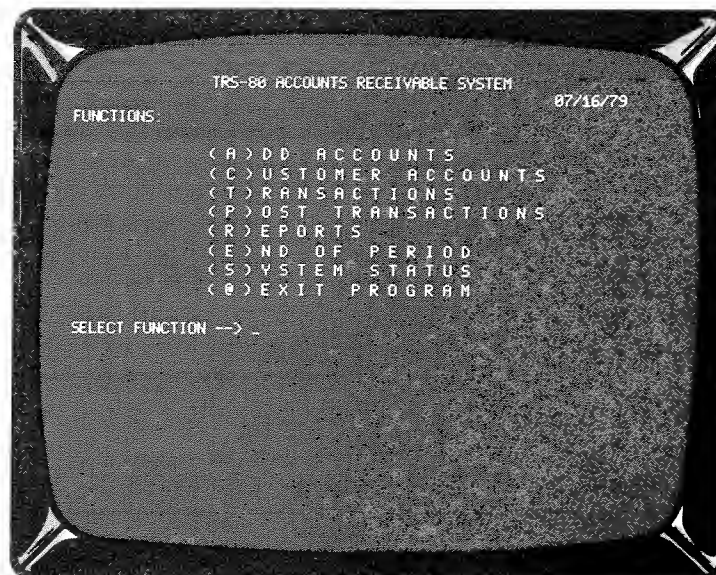
Type **Y** or **N**. If you type **Y** (Yes), the system will ask you to enter your password. Type in your password carefully. Press **ENTER** after you type in the last letter. A # symbol will appear on the screen for each character of the password.

If you make a mistake in typing, the computer will display:

\*\*\*INVALID PASSWORD\*\*\*  
PROGRAM TERMINATED

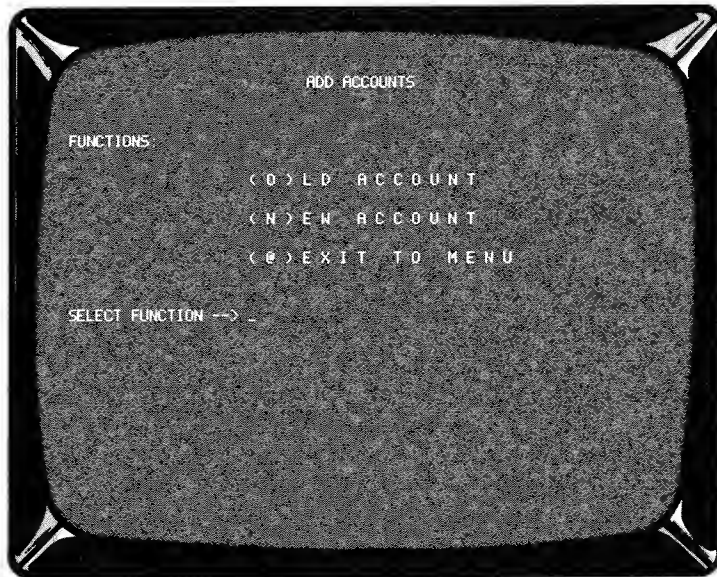
Type **RUN**, and press **ENTER**. The system will display the date and the "Date Correct (Y/N)" question. Press the **Y** key and the system will ask you again for the password. Type it in carefully.

When the password is correct, the program will search for all the stored information and display the Main Menu:



You select a function by pressing the appropriate key (the letter in parenthesis). We want to Add our accounts now, so press **A** for (A)dd Accounts.

The screen will show:



#### (O)ld Account

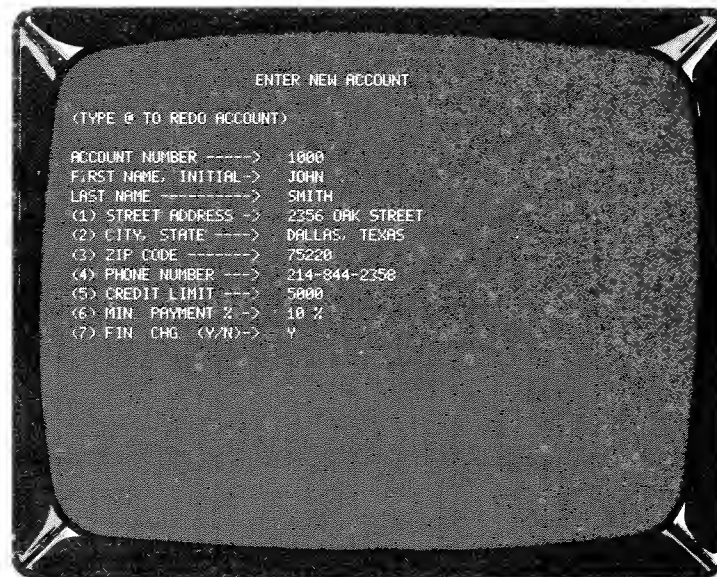
The "Old Account" option is used to transfer existing customer accounts into ARS. This option allows you to enter all the pertinent account information including: the account's previous balance, amount and date of last sale, amount and date of last payment, and the present credit status of the account.

#### (N)ew Account

The "New Account" option is for entering new customers that do not have a credit history with your company. All of the pertinent information is stored with the exception of previous balance, sales, payments, and current credit status.

Let's enter a New Account first, so press **[N]**.

The first “page” of a customer record will appear on the screen:



```

                                ENTER NEW ACCOUNT

      (TYPE @ TO REDO ACCOUNT)

ACCOUNT NUMBER ----> 1000
FIRST NAME, INITIAL-> JOHN
LAST NAME -----> SMITH
(1) STREET ADDRESS-> 2356 OAK STREET
(2) CITY, STATE ---> DALLAS, TEXAS
(3) ZIP CODE -----> 75220
(4) PHONE NUMBER ---> 214-844-2358
(5) CREDIT LIMIT ---> 5000
(6) MIN. PAYMENT %-> 10 %
(7) FIN. CHG. (Y/N)-> Y

```

In the example above, a typical account was used to show how your information should appear. Press **ENTER** after each line you type. If you make a mistake, press the **@** key to redo the entire entry. Lines 1 through 5 are pretty much self-explanatory.

For Line 6, you indicate the minimum payment the customer may remit. This is a percentage of the total amount owed by the customer. Type the number without the percent sign. For example: If you want the customer to pay 10% of his total bill each month, type **10** and press **ENTER**. ARS will show this as 10%.

Line 7 is a question:

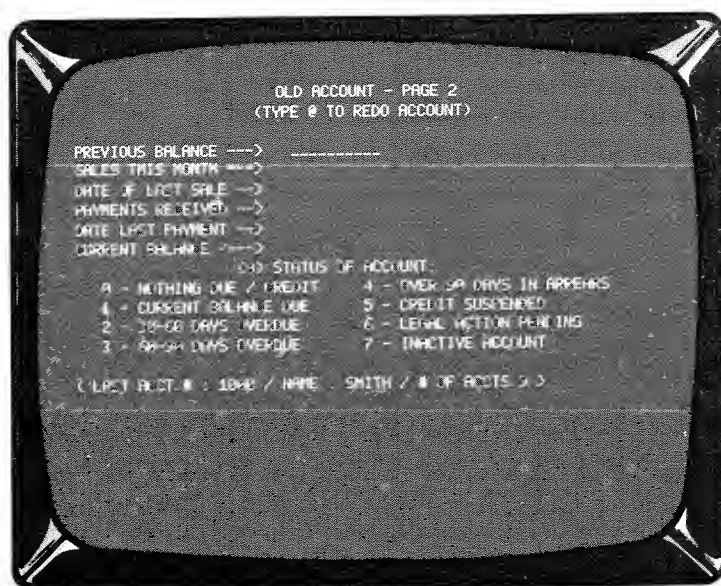
DO YOU WISH TO APPLY THE FINANCE CHARGE? (Y/N)

Answer **Y** or **N**. That's the last information the program needs about a “new” account. ARS will store the account on the data disk.

Now let's try an Old Account. Press **Ⓢ** to return to the Add Accounts Menu, then press **Ⓟ**. The first "page"\* of the customer account will appear on the screen with the heading:

ENTER OLD ACCOUNT

\*Note: There are two "pages" to an "old account". When you're through with the first "page" the TRS-80 will automatically display the second "page":



When you type in numbers, enter all dollar values with a decimal point, followed by two digits. For example: **120.00**. Enter the date in the **MM/DD/YY** form. For example: **04/04/79**. If you don't enter information correctly, the program will not accept your entries. You **must** enter some dollar value, even if it is just **0.00**. If a date entry is not necessary, simply press **ENTER**.

When you've finished, the computer will automatically compute the Current Balance, then display the results.

Next, you need to enter a "status" code number. Every code number on the "(8) STATUS OF ACCOUNT" table represents an Account Status "type". Enter the correct number for the current status of the account. The TRS-80 will then store the account.

**Note:** If you have an "Inactive Account" (status code 7), the account will not appear on billing statements.

When you've entered all of your accounts, type @ to exit to the Main Menu.

The computer will sort the accounts three ways:

1. By Account Number
2. Alphabetically, by Last Name
3. By Zip Code

**Note:** It will take about 15 minutes to sort 300 accounts and about 25 minutes to sort 500 accounts.

ARS will check to see if any duplicate account numbers have been entered. If you have duplicates, record the account numbers and delete the duplicates before doing any transactions.

Press **ENTER** to return to the Main Menu.

**(C)ustomer Accounts**

This function lets you find any Customer Account for inspection, editing, or printing. At the Main Menu, press **[C]**.

The screen will show:

ACCESS BY: (A)CCOUNT NUMBER OR BY (N)AME?

If you want the account by Account Number, press **[A]**. If by the Customer's Name, press **[N]**.

If you press **[A]**, type the account number and press **[ENTER]**. The requested account will appear on the screen.

If you press **[N]**, type the customer's last name and press **[ENTER]**. If you press **[@]**, you will return to the Main Menu.

In additions to the customer information (name, address, etc.), you'll see a table of options:

(S)EARCH	(P)REVIOUS	(N)EXT	(T)URN PAGE
(U)PDATE	(H)ARD COPY	(D)ELETE	(@) EXIT TO MENU

This is what each option does:

**(S)earch**

Search will instruct the TRS-80 to Search for an account you want. The computer will ask:

ACCESS BY: (A)CCOUNT NUMBER OR BY (N)AME?

Press either **[A]** or **[N]**, then type the Account Number, or the Name of the Customer. When you've entered the information, the desired account will appear on the screen.

### (P)revious

Previous steps through the accounts backward. When you press **P**, the computer will display the account “in front of” the account you’ve just seen.

### (N)ext

When you press **N**, the computer will display the Next Account in the record. If no additional accounts appear, you’ve reached the last account.

### (T)urn Page

A customer’s account consists of two “pages”. In all of the options we’ve discussed, the program brings only the first page to the screen. To see the second page, you must use the Turn Page option. Press **T** after you’ve seen the first page. If you need to go back to the first page, press **T** again.

### (U)pdate

Uppdate is used to edit account information. The option table will be replaced by:

SELECT LINE NUMBER TO BE CHANGED

This applies to each “line” of the customer’s account record, such as, street address, etc. Type the number (of the line) which needs editing, make the desired changes, and press **ENTER**. The computer will display the corrected line.

If you change a zip code, the ARS will sort the accounts again. The sorting occurs when you return to the Main Menu.

**Note:** You can only make changes in registered dollar amounts by entering a new transaction in the system.

### (H)ard Copy

Hard Copy lets you print any account in the record. First, use the (S)earch option to bring the desired account to the screen. Make sure the line printer is on. Press **[H]**. The printer will print both pages of the account you've selected.

### (D)elete

Delete allows you to completely eliminate an account from the system. First, use the (S)earch option to bring the desired account to the screen, then press **[D]** to Delete the account.

The screen will show:

OK TO DELETE? (Y/N)

**Caution:** Once an account is deleted, it is gone for good. You can't get it back. After you delete an account, a "Bad Debt" entry is automatically registered in the Transaction File. This entry is a key to the amount owed by the customer (at the time of deletion) and is entered for the benefit of the auditor.

Now that we have finished storing and editing the accounts, let's go to the Main Menu, and enter some typical transactions. Press the **[@]** key.

### (T)ransactions

Customer transactions are entered with this function. Press **T**.

The screen will show:

TRANSACTIONS

The display will remain on the screen for a few seconds while the computer finds the program and loads it into memory.

When the computer has the program ready, the screen will show:

(N)ORMAL OR (C)ORRECTION ENTRY?

Unless you wish to correct previous entries, type **N**.

The screen will show:

(S)ALES, (P)AYMENT OR (B)AD DEBT?

If you press **S**, the computer will ask for the customer's Account Number. Type the account number and press **ENTER**. The computer will check for a valid account number.

The TRS-80 will then ask you to enter the date.\* Use the standard MM/DD/YY form and press **ENTER**.

\*Note: If you want to use today's date, simply press **ENTER**. (Today's date is the date you typed at the beginning of the ARS program.)

After the date entry, the computer will ask for:

CUSTOMER INVOICE NUMBER

Type the invoice number and press **ENTER**.

The screen will show:

GLC:\_\_

The program is asking for one of the General Ledger Codes you specified during the setup procedure. Type the GLC number and press **ENTER**. The computer will display the five character General Ledger Description, then ask for the amount. Type in the amount (including a decimal point and 2 digits). Press **ENTER**. This is how your transaction entry will appear:

GLC: 5 SALES AMOUNT: 50.00

The ARS will ask for a total of three General Ledger Code entries. You can use less than 3, if you wish. (Skip over any unwanted GLC requests by pressing **ENTER**.)

At this point, the TRS-80 will compute the amount (or amounts) you entered. The computer will display the total amount, record the transaction and display the transaction as the last entry recorded.

The program is now ready for another transaction.

#### (P)ayment

The Payment process is similar to (S)ale, except that you'll be asked for only one GLC number. This should be the General Ledger Code assigned for Cash (or any other account you designate to record incoming payments). A completed line for Payment will appear on the screen. For example:

GLC: 8 CASH AMOUNT OF PAYMENT: 50.00

#### (B)ad Debt

The Bad Debt Transaction process requires only the account number, date and reference number be entered. Use the invoice number line to enter a reference number as an audit trail. The completed line for a bad debt will look similar to this:

GLC: 4 B/D AMOUNT OF BAD DEBT: 300.00

Be very careful when entering the account number. Once you complete the Bad Debt process, the account's balance will be removed from the total of receivables and the account will be placed on "Credit Suspended" status.

## How To Make Correction Entries

The screen will show:

(N)ORMAL OR (C)ORRECTION ENTRY?

Select Correction Entry. Go through the same procedures as Normal Entry. The correction entry will replace the previous entry. Corrections may be made at any time before "End of Period Processing".

Make sure that the customer account and invoice number (of the original, and correcting transactions) are **exactly the same** during your correction procedure. If you fail to correct a mistake, the errors in the file are compounded and you may damage still another account.

## Crediting A Customer's Account For Returns

There are times when it is necessary to accept merchandise as a return for credit, or to offer an allowance as a special discount for damaged or defective merchandise. To credit a customer's account, prepare a Normal Sales Entry, and enter the amount as a **negative** number. (You must have a GLC code for "Returns".) The customer's statement will show a negative charge and your Posting Report will show a negative sale.

## (P)ost Transactions

It is at this point that all the transactions (from the Transaction File) are debited or credited to the appropriate customer accounts. At the same time, amounts for each of the General Ledger accounts are totaled and shown as debits/credits to the General Ledger's "Accounts Receivable" account.

Postings should be made at frequent intervals: daily, biweekly, or weekly. End of Period Processing cannot take place if there are unposted transactions in the system.

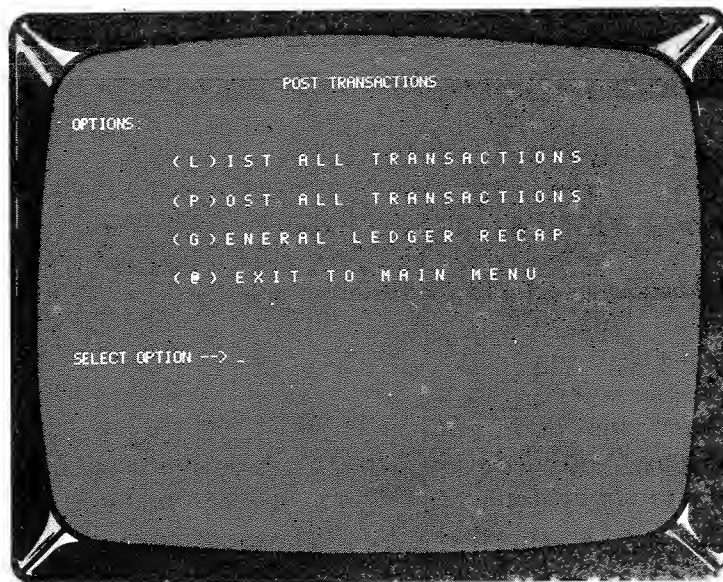
One of two possible displays will appear after you press **[P]**.

1. If the system has no outstanding transactions to post, the screen will show:

TRANSACTION FILE IS EMPTY

The program will automatically return to the Main Menu.

2. If transactions have been entered (using the Transactions Option), the screen will show:



Reminder: This function requires printing. Be sure the line printer is on.

### (L)ist All Transactions

Press **[L]** to print the entire Transaction File. The printout will contain full information on the status and type of each transaction — posted or unposted, sales or payment, normal or correction entry. Additional features are:

1. Prints transactions, using the respective GL code numbers.
2. Prints the amounts of each transaction, including totals.
3. Prints invoice numbers and account numbers for each transaction.

### (P)ost All Transactions

Press **[P]** to Post all transactions and print a serialized report (which documents the posting). The first part of this report is similar to the Transaction File printout, with one difference. The first part of the Posting Report contains the customer names.

The second part of the report lists the totals to be applied to the various General Ledger Accounts, such as: Sales, Sales Tax, Shipping Charges, etc.

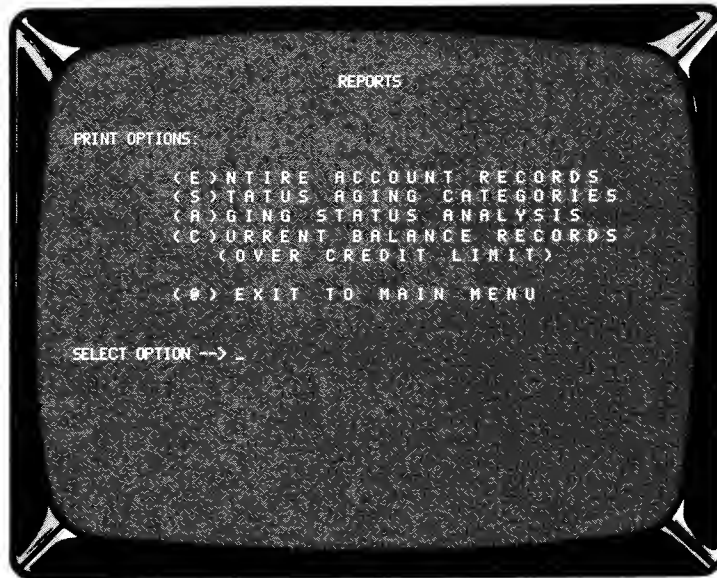
Posting is a very important function, since all transactions must be recorded to the proper customer's account. It also takes the subtotals of each General Ledger Account and adds them to the Accumulated Totals-to-Date.

### (G)eneral Ledger Recap

This function provides a General Ledger Account Activity Report, which is similar in format to the second part of the Posting Report. The Recap Report contains the accumulated totals for the current month, grouped by General Ledger Codes.

## (R)eports

(R)eports will print reports which cover particular functions. When you press **R**, the screen will show:



### (E)ntire Account Records

You select this option by pressing **E**. The program then asks if you wish the account printed by:

(A)CCOUNT #, (N)AME, or (Z)IP CODE SEQUENCE?

Whether you type **A**, **N**, or **Z**, the next display will be:

(A)BBREVIATED OR (F)ULL RECORDS?

Examples of these reports are shown in Appendix E (on Pages 42-44). At the bottom of each printout is the total amount of receivables for the records and the percentage of total receivables the amount represents.

The printer will print the report when you press **A** or **F**.

### (S)tatus Aging Categories

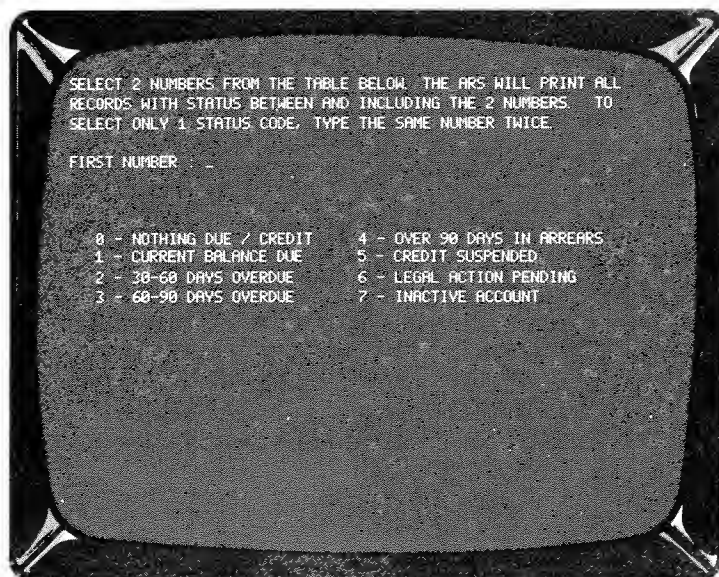
You use this option to print accounts by aging status. The report will tell you how long accounts have stayed unpaid. Press **[S]**. The computer will ask if you wish the aging status listing printed by:

(A)CCOUNT #, (N)AME, or (Z)IP CODE SEQUENCE

When you have typed your answer, the computer will ask:

(A)BBREVIATED OR (F)ULL RECORDS

Following your answer, the screen will show:



For example: If you want all accounts 90 days in arrears, credit suspended, and legal action pending, type **[5][7][6]**. For all accounts with nothing due, type **[0][0]**.

The screen will show:

OK TO PRINT? (Y/N)

Press **[Y]** if you're ready to print a Status Aging Report.

**(A)ging Status Analysis**

When you press **[A]**, the screen will show:

## SCANNING ACCOUNT RECORDS

After all account records have been scanned and summed by the computer, you'll see a display of results on the screen with a question:

DO YOU WANT A PRINTOUT? (Y/N)

Type **[Y]** if you want a printed copy. The "analysis" printout summarizes the aging categories of all account records. Below is a sample of a typical analysis:

AGING CATEGORY	ACCTS	AMOUNT	% OF A/R
0 - NOTHING DUE / CREDIT	4	47.23+	1.48 %
1 - CURRENT BALANCE DUE	58	1250.25+	39.08 %
2 - 30-60 DAYS OVERDUE	27	882.59+	25.08 %
3 - 60-90 DAYS OVERDUE	13	523.63+	16.37 %
4 - OVER 90 DAYS IN ARREARS	10	363.21+	11.35 %
5 - CREDIT SUSPENDED	4	125.36+	3.92 %
6 - LEGAL ACTION PENDING	1	32.48+	1.02 %
7 - INACTIVE ACCOUNT	2	54.74+	1.71 %
TOTALS	111	3199.49+	100.00 %

DO YOU WANT A PRINTOUT ? (Y/N) \_

**(C)urrent Balance Records (Over Credit Limit)**

Press **[C]** to print all accounts with a current balance over the credit limit. Before printing, you must instruct the computer if you wish to print by (A)ccount #, (N)ame, or (Z)ip Code Sequence, and in (A)bbreviated, or (F)ull form.

### **(E)nd of Period Processing**

This function is used at the end of a billing period to:

1. Print the monthly billing statements.
2. Update all customer accounts.
3. Clear the system to let you enter next month's transactions.
4. Add all applicable finance charges.

### **Billing Statement Message**

Press the **[E]** key.

The screen will show:

ENTER THIS MONTH'S MESSAGE (2 LINES):  
1ST LINE:  
2ND LINE:

This allows you to enter an advertising message, which will appear at the bottom of each billing statement. You have two lines, of 41 characters each, for this message. You can skip one, or both of these lines (if your firm does not want a message), by pressing **[ENTER]**.

### **Period Ending Date**

If you've typed in a message, press **[ENTER]**.

The screen will show:

PERIOD ENDING DATE (PRESS<ENTER>FOR TODAY'S DATE):

If the ending date of the billing cycle is the same date you entered at the start of the program, press **[ENTER]**. If the two dates are different, type in the ending date of the billing cycle and press **[ENTER]**.

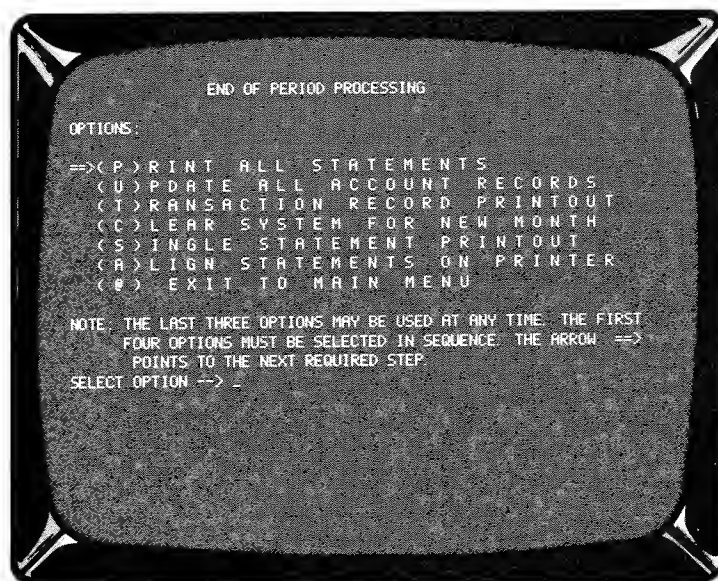
The period ending date will appear on the billing statements. A word of caution: Never enter any transactions that occur after the period ending date. Enter those transactions the following month.

The screen will show:

BILLING DATE (PRESS <ENTER> FOR TODAY'S DATE):

Type the date that you want to appear on your statement and press **ENTER**.

The screen will show:



Please Note:

1. The first four options must be selected in order (if you wish to do End of Period Processing).
2. The last three options may be used at any time. (You'll note that an arrow points to the first option in the menu. This arrow will move down to the second option after the first has been used. It will advance until you get to the fifth option.)

Here's a good system to use for End of Period Processing:

1. Align printer for statements.
2. Print all statements.
3. Update all account records.
4. Print a Transaction Record Report.
5. Clear system for new month.

### (A)lign Printer For Statements

The first step in the billing process is to load and align the printing forms. Press **[A]** for this option. This is what you'll see on the screen.

(H)ORIZONTAL ALIGNMENT TEST  
(V)ERTICAL ALIGNMENT TEST  
(A)LIGNMENT TEST FINISHED

Press **[H]** for Horizontal Adjustment of billing forms. This will start the printer which will print this message:

CENTER THE X'S IN THE PERFORATIONS

If pre-printed billing forms are being used, make sure that the vertical perforations on the forms are aligned to lie in the middle of the printed X's. (See Appendix D in the back of the manual.) The knob for adjusting the length of the printed line is at the back of the line printer.

Now, press **[V]** for Vertical Adjustment. This function will execute a top of form. Adjust the form vertically, as illustrated. Be sure that the bottom of the printer's ruler is lined up on the form as shown. When you're finished, press **[A]** for Alignment Test Finished.

**(P)rint All Statements**

This option is used to calculate the amount owed by each customer (including finance charge). It will print statements for all active accounts with a non-zero balance.

Press **[P]**. The program will ask if the statements should be printed by:

(A)CCOUNT NUMBER, (N)AME, or (Z)IP CODE

Enter your selection. Information in the printed statements will include: (a) previous balance, (b) all sales transactions, (c) payments, (d) finance charges on the previous month's unpaid balance minus any payments received, (e) total due, and (f) minimum payment. The account's aging status is printed under the customer's address.

A single billing form can list 25 transactions. If a customer had more than 25 transactions during a particular billing period, the account statement will continue on succeeding pages. At the bottom of a page, the words "Continued on Next Page" will indicate a continuation. On the next page, the "Previous Balance" space will show the balance amount (carried over from the preceeding page). This method is used (in succeeding pages) to accommodate up to 255 transactions per customer for a single billing period.

The monthly messages and totals will be printed on the final page of the statement.

**The screen will show:**

DO YOU APPROVE OF THE STATEMENTS AS PRINTED? (Y/N)

Examine the printout for errors or problems with forms. If you find errors or damage to forms, press **[N]**. \* You can now reprint all the statements by pressing **[P]**. If the printout meets your approval this time, press **[Y]**. Do not exit (using @) unless you have approved the statements.

**\*Note:** If just one statement needs reprinting, press **[Y]**. You can make a Single Statement Printout afterwards.

## The Rest Of The Billing Options

### (U)pdate

Before you Udate, you must have all billing statements in order. Correct any statements that are wrong or misprinted. You can leave “End of Period Processing” at anytime, make corrections, and then return to the point from which you left. The arrow will point to the next function that must be used to complete the End of the Period Processing. Now you can press U to Udate all account records automatically.

When Updating is finished, press T for a printed copy of the End Of Period Transaction File — you’ll need this for an audit trail. Afterwards, press C to Clear the system for a new billing period.

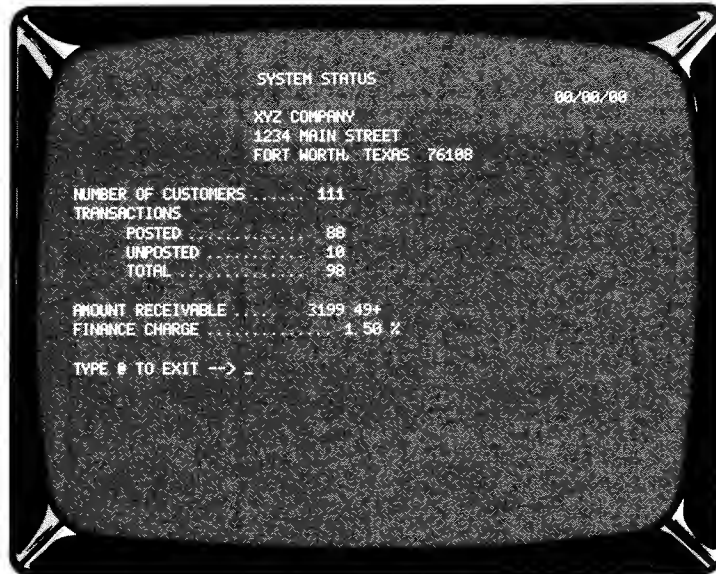
### (S)ingle Statement Printout

For a Single Statement printout, type S. The program will ask for the number of the Account. When you’ve typed the number, the statement will be printed.

When you’re finished with the Billing Section, press @ to return to the Main Menu.

## (S)ystem Status

You can use this function at any time. It lets you inspect the current system status at a glance. When you press **[S]**, a display (similar to the one below) will appear:



## (@) Exit Program

Type **[E]** to Exit from the program. Always use this command when you're finished using ARS. This function automatically closes any files that might be open and makes sure that all changes have been properly recorded.

## The @ Key (In General)

The @ key may be used either to restart a function or to Exit to the Main Menu. If you're entering data and you make a mistake, press **[@]** to restart the function.

ARS uses the @ key as an eraser for a variety of mistakes. Whenever it is easier to start over, rather than correct a number of errors, use the @ key. Pressing the @ key several times will take you back to the start of the program quickly.

## The ← Key

If you make a mistake while typing a line, use the ← key to backspace to the error. The ← key is directly above the **[ENTER]** key. Hold down the SHIFT key and press the ← key to erase a long line.



## Appendix A – How To Format Your Data Diskettes

This process prepares blank diskettes for use on the disk system. All data diskettes must be formatted before being used. Here's how you do it:

1 — Insert a blank diskette in Drive #1

2 — Do the following exactly as shown:

### Screen Shows:

DOS READY  
WHICH DRIVE etc.  
DISKETTE NAME?  
CREATION DATE?  
MASTER PASSWORD  
LOCK OUT ANY TRACKS?  
HIT "ENTER" TO CONTINUE

### You Type:

<b>F</b>	<b>O</b>	<b>R</b>	<b>M</b>	<b>A</b>	<b>T</b>	press	<b>ENTER</b>
<b>1</b>						press	<b>ENTER</b>
<b>D</b>	<b>A</b>	<b>T</b>	<b>A</b>	<b>7</b>	<b>9</b>	press	<b>ENTER</b>
<b>0</b>	<b>1</b>	<b>/</b>	<b>0</b>	<b>1</b>	<b>/</b>	<b>7</b>	<b>9</b>
<b>P</b>	<b>A</b>	<b>S</b>	<b>S</b>	<b>W</b>	<b>O</b>	<b>R</b>	<b>D</b>
<b>N</b>	<b>O</b>						
						press	<b>ENTER</b>
						press	<b>ENTER</b>

## Appendix B – How To Backup Your Diskettes

**Note:** In order to copy diskettes, you must have a diskette that contains the BACKUP command. You can use the TRSDOS 2.2 Diskette (supplied with Drive 0). Once you load the BACKUP function into the computer, the disk may be replaced with the disk you want to copy.

Use this procedure **EXACTLY**:

1. Turn on everything except the TRS-80 keyboard. If this is the first time you've ever used the Radio Shack Disk System, refer to the Disk Operating System Manual for detailed instructions.
2. Insert a new, blank diskette in Drive 1 (farthest from the Interface) with the diskette notch up, label to the right.
3. Close the Drive 1 door.
4. Insert the TRSDOS 2.2 diskette in Drive 0 with the notch up, label facing right, and close the door. (Drive 0 is the first Drive connected to the Interface.)
5. Turn on the TRS-80 keyboard. (The switch on the right rear apron.)

The screen will show:

DOS READY

You will type:

**B A C K U P** and press **ENTER**

Wait until the red light on the Disk Drive goes off. Remove the TRSDOS 2.2 diskette from Drive 0, and place it in its protective sleeve. Insert the diskette you wish to copy in Drive 0 and close the door.

The screen will show:

SOURCE DRIVE?  
DESTINATION DRIVE?\_  
BACKUP DATE (MM/DD/YY)

You will type:

**0** and press **ENTER**  
**1** and press **ENTER**  
Today's Date and press **ENTER**

The computer will format the disk and transfer the data to the blank diskette. When it's finished, the screen will show:

HIT 'ENTER' TO CONTINUE

Don't press the **ENTER** key just yet.

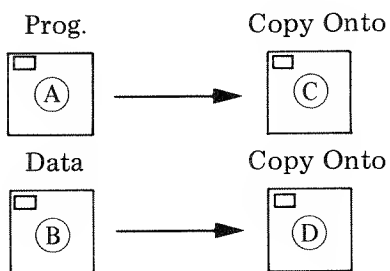
Remove the copy diskette from Drive 1 and place it in its protective sleeve. Remove the diskette from Drive 0. You can now replace the TRSDOS 2.2 diskette in Drive 0, close the door, and press **ENTER** to continue.

When you're using your system, get in the habit of making copies of both diskettes at the end of the day. The next day, use the copies that you just made. It's a very inexpensive form of insurance. You might consider even carrying the backup copies home at night in case of fire, theft, or vandalism. You must also save the diskettes that contain the last completed month.

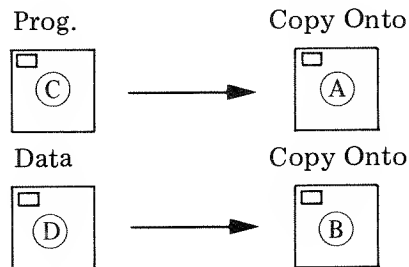
Here's an example, using two extra diskettes:

The first day's diskettes we will call A and B. The next day's diskettes will be called C and D.

### Day One



### Day Two



On the second day, you will take A and B, and use them as blank diskettes to copy C and D. If something happens to a diskette, you still have the last backup copy to rely on. Four extra diskettes will give you three days of record protection.

Be sure to keep the last **completed** month's set of diskettes till you complete the current month. This is very important in case of a major problem.

**Note:** The most effective method for erasing diskettes is by using a Radio Shack Bulk Tape Eraser, Stock No. 44-210.

## Appendix D – Statement Alignment Mask

[illegible]

## Complete Transaction File

07/17/79: COMPLETE TRANSACTION FILE

TYPE	ACCT#	NAME	DATE	INV#	REMARKS
	AMOUNT (GLC)	AMOUNT (GLC)	AMOUNT (GLC)	TOTAL (GLC)	
SALE	1000	07/17/79	1	NORMAL-UNPOSTED	
*	25.25+( 5 )	12.25+( 7 )	1.87+( 9 )	39.37+( 1 )	
SALE	1001	07/12/79	2	NORMAL-UNPOSTED	
*	369.24+( 6 )	0.00+( 0 )	0.00+( 0 )	369.24+( 1 )	
SALE	1002	07/13/79	3	NORMAL-UNPOSTED	
*	124.78+( 5 )	12.98+( 8 )	25.23+( 10 )	162.99+( 1 )	
SALE	1003	07/17/79	4	NORMAL-UNPOSTED	
*	118.45+( 5 )	0.00+( 0 )	0.00+( 0 )	118.45+( 1 )	
PMNT	1004	07/20/79	5	NORMAL-UNPOSTED	
*	100.00-( 7 )	0.00+( 0 )	0.00+( 0 )	100.00-( 2 )	
SALE	1005	07/17/79	6	NORMAL-UNPOSTED	
*	254.21+( 5 )	23.23+( 7 )	39.35+( 10 )	316.79+( 1 )	
SALE	1006	07/21/79	7	NORMAL-UNPOSTED	
*	1425.31+( 6 )	114.02+( 8 )	0.00+( 0 )	1539.33+( 1 )	
PMNT	1007	07/17/79	8	NORMAL-UNPOSTED	
*	25.00-( 7 )	0.00+( 0 )	0.00+( 0 )	25.00-( 2 )	
SALE	1008	07/17/79	9	NORMAL-UNPOSTED	
*	502.45+( 5 )	25.23+( 11 )	0.00+( 0 )	527.68+( 1 )	
SALE	1009	07/25/79	9	NORMAL-UNPOSTED	
*	123.45+( 6 )	9.87+( 8 )	0.00+( 0 )	133.32+( 1 )	
SALE	1003	07/25/79	10	NORMAL-UNPOSTED	
*	245.69+( 5 )	17.69+( 8 )	30.00+( 11 )	293.38+( 1 )	

END OF LISTING

# Sample Reports (continued)

## Posting Report

07/17/79 - POSTING REPORT # 1

TYPE	ACCT#	NAME	DATE	INV#	REMARKS
	AMOUNT (GLC)	AMOUNT (GLC)	AMOUNT (GLC)	TOTAL (GLC)	
SALE	1000	ADAMS	07/17/79	1	NORMAL-POSTED
*	25.25+( 5 )	12.25+( 7 )	1.87+( 9 )	39.37+( 1 )	
SALE	1001	WASHINGTON	07/12/79	2	NORMAL-POSTED
*	369.24+( 6 )	0.00+( 0 )	0.00+( 0 )	369.24+( 1 )	
SALE	1002	JEFFERSON	07/13/79	3	NORMAL-POSTED
*	124.78+( 5 )	12.98+( 8 )	25.23+( 10 )	162.99+( 1 )	
SALE	1003	TAFT	07/17/79	4	NORMAL-POSTED
*	118.45+( 5 )	0.00+( 0 )	0.00+( 0 )	118.45+( 1 )	
PMNT	1004	LINCOLN	07/20/79	5	NORMAL-POSTED
*	100.00-( 7 )	0.00+( 0 )	0.00+( 0 )	100.00-( 2 )	
SALE	1005	TRUMAN	07/17/79	6	NORMAL-POSTED
*	254.21+( 5 )	23.23+( 7 )	39.35+( 10 )	316.79+( 1 )	
SALE	1006	FAIRLEY	07/21/79	7	NORMAL-POSTED
*	1425.31+( 6 )	114.02+( 8 )	0.00+( 0 )	1539.33+( 1 )	
PMNT	1007	HANCOCK	07/17/79	8	NORMAL-POSTED
*	25.00-( 7 )	0.00+( 0 )	0.00+( 0 )	25.00-( 2 )	
SALE	1008	MADISON	07/17/79	9	NORMAL-POSTED
*	502.45+( 5 )	25.23+( 11 )	0.00+( 0 )	527.68+( 1 )	
SALE	1009	FORD	07/25/79	9	NORMAL-POSTED
*	123.45+( 6 )	9.87+( 8 )	0.00+( 0 )	133.32+( 1 )	
SALE	1003	TAFT	07/25/79	10	NORMAL-POSTED
*	245.69+( 5 )	17.69+( 8 )	30.00+( 11 )	293.38+( 1 )	

GENERAL LEDGER INPUTS

	DEBIT(+)		CREDIT(-)
	-----		-----
GLC: 3 1000 / A/R	0.00+	1500 / F/C	0.00+
GLC: 4 1000 / A/R	0.00+	2000 / B/D	0.00+
GLC: 5 1000 / A/R	1270.83+	2500 / CASH1	1270.83-
GLC: 6 1000 / A/R	1918.00+	2501 / CASH2	1918.00-
GLC: 7 2502 / CASH3	89.52+	1000 / A/R	89.52-
GLC: 8 1000 / A/R	154.56+	3000 / SLTAX	154.56-
GLC: 9 1000 / A/R	1.87+	3001 / FDTAX	1.87-
GLC: 10 1000 / A/R	64.58+	3500 / FRGHT	64.58-
GLC: 11 1000 / A/R	55.23+	4000 / MISC.	55.23-

END OF REPORT

General Ledger Recap

07/17/79 - GENERAL LEDGER RECAP (TOTALS THIS MONTH TO DATE) - REPORT # 1

	DEBIT(+)		CREDIT(-)
	-----		-----
GLC: 3 1000 / A/R	0.00+	1500 / F/C	0.00+
GLC: 4 1000 / A/R	0.00+	2000 / B/D	0.00+
GLC: 5 1000 / A/R	1270.83+	2500 / CASH1	1270.83-
GLC: 6 1000 / A/R	1918.00+	2501 / CASH2	1918.00-
GLC: 7 2502 / CASH3	89.52+	1000 / A/R	89.52-
GLC: 8 1000 / A/R	154.56+	3000 / SLTAX	154.56-
GLC: 9 1000 / A/R	1.87+	3001 / FDTAX	1.87-
GLC: 10 1000 / A/R	64.58+	3500 / FRGHT	64.58-
GLC: 11 1000 / A/R	55.23+	4000 / MISC.	55.23-

END OF REPORT

# Complete Account Listing – Run #1

07/17/79 - COMPLETE ACCOUNT LISTING - RUN# 1

(ACCT# SEQUENCE)

ACCT#	LAST NAME	CRT LMT	PRV BAL	TTL SLS	PMT REC	CUR BAL	STATUS
1000	ADAMS	5000	0.00+	39.37+	0.00+	39.37+	0
1001	WASHINGTON	5000	0.00+	369.24+	0.00+	369.24+	0
1002	JEFFERSON	5000	0.00+	162.99+	0.00+	162.99+	0
1003	TAFT	1000	0.00+	411.83+	0.00+	411.83+	0
1004	LINCOLN	1000	0.00+	0.00+	100.00-	100.00-	0
1005	TRUMAN	1500	0.00+	316.79+	0.00+	316.79+	0
1006	FAIRLEY	1000	0.00+	1539.33+	0.00+	1539.33+	0
1007	HANNCOCK	500	0.00+	0.00+	25.00-	25.00-	0
1008	MADISON	1000	0.00+	527.68+	0.00+	527.68+	0
1009	FORD	1500	0.00+	133.32+	0.00+	133.32+	0

RECEIVABLES FOR THESE RECORDS: 3375.55+

THIS IS 100.00 % OF 3375.55+ (TOTAL RECEIVABLES).

END OF REPORT

## Complete Account Listing – Run #2.

07/17/79 - COMPLETE ACCOUNT LISTING - RUN# 2

(NAME SEQUENCE)

ACCT#	NAME OF ACCOUNT	ADDRESS	CITY, STATE	ZIP	
PHONE	CRT LMT	%RQ	FC?	DT LAST SALE	DT LAST PYMT
PRV BAL	TTL SLE	PMT RCV	CUR BAL	STATUS	
1000	JOHN ADAMS	3698 OAK STREET	NEW YORK, N. Y.	10016	
212-257-9853	5000	10	Y	07/17/79	00/00/00
0.00+	39.37+	0.00+	39.37+	0 - NOTHING DUE / CREDIT	
1006	BILLIE FAIRLEY	14 RIVERVIEW RD	TARRYTOWN, N. Y.	10591	
914-769-7246	1000	10	N	07/21/79	00/00/00
0.00+	1539.33+	0.00+	1539.33+	0 - NOTHING DUE / CREDIT	
1009	GERALD FORD	2135 ELM STREET	OAKLAND, CA	90045	
203-456-7890	1500	10	Y	07/25/79	00/00/00
0.00+	133.32+	0.00+	133.32+	0 - NOTHING DUE / CREDIT	
1007	JOHN HANCOCK	9638 MADISON AVE	NEW YORK, N. Y.	10021	
212-321-1596	500	10	Y	00/00/00	07/17/79
0.00+	0.00+	25.00-	25.00-	0 - NOTHING DUE / CREDIT	
1002	THOMAS JEFFERSON	8712 EAST 34TH ST	BROOKLYN, N. Y.	11211	
212-998-5442	5000	10	Y	07/13/79	00/00/00
0.00+	162.99+	0.00+	162.99+	0 - NOTHING DUE / CREDIT	
1004	ABRAHAM LINCOLN	235 PICK AVE	JAMAICA, N. Y.	11476	
212-425-7963	1000	10	Y	00/00/00	07/20/79
0.00+	0.00+	100.00-	100.00-	0 - NOTHING DUE / CREDIT	
1008	JAMES MADISON	2562 HIGH BLUFF	MONTICELLO, NY	11023	
914-325-7916	1000	10	Y	07/17/79	00/00/00
0.00+	527.68+	0.00+	527.68+	0 - NOTHING DUE / CREDIT	

## Sample Reports (continued)

### Complete Account Listing – Run #2 (continued)

ACCT#	NAME OF ACCOUNT	ADDRESS	CITY, STATE	ZIP	
PHONE	CRT LMT	%RQ	FC?	DT LAST SALE	DT LAST PYMT
PRV BAL	TTL SLE	PMT RCV	CUR BAL	STATUS	
1003	WILLIAM TAFT	243-59 43RD AVE.	FOREST HILLS, NY	11202	
212-654-3247	1000	10	Y	07/25/79	00/00/00
0.00+	411.83+	0.00+	411.83+	0 - NOTHING DUE / CREDIT	
1005	HARRY TRUMAN	4683 WEST 74TH ST	NEW YORK, N. Y.	10036	
212-639-4587	1500	10	Y	07/17/79	00/00/00
0.00+	316.79+	0.00+	316.79+	0 - NOTHING DUE / CREDIT	
1001	GEORGE WASHINGTON	2354 CHERRY LANE	FORT WORTH, TX	76102	
817-844-3697	5000	10	Y	07/12/79	00/00/00
0.00+	369.24+	0.00+	369.24+	0 - NOTHING DUE / CREDIT	

RECEIVABLES FOR THESE RECORDS: 3375.55+  
 THIS IS 100.00 % OF 3375.55+ (TOTAL RECEIVABLES).

END OF REPORT

## Listing By Aging Status

07/17/79 - LISTING BY AGING STATUS - RUN# 1

1 - CURRENT BALANCE DUE  
TO  
4 - OVER 90 DAYS IN ARREARS

&lt;ACCT# SEQUENCE&gt;

ACCT#	LAST NAME	CRT LMT	PRY BAL	TTL SLS	PMT REC	CUR BAL	STATUS
1003	TAFT	1000	0.00+	411.83+	0.00+	411.83+	1
1005	TRUMAN	1500	0.00+	316.79+	0.00+	316.79+	3
1006	FAIRLEY	1000	0.00+	1539.33+	0.00+	1539.33+	4
1007	HANNCOCK	500	0.00+	0.00+	25.00-	25.00-	2

RECEIVABLES FOR THESE RECORDS: 2242.95+  
THIS IS 66.45 % OF 3375.55+ (TOTAL RECEIVABLES).

END OF REPORT

## Accounts Receivable Analysis By Aging Status

---

### ACCOUNTS RECEIVABLE ANALYSIS BY AGING STATUS

AGING CATEGORY	ACCTS	AMOUNT	% OF A/R
-----	-----	-----	-----
0 - NOTHING DUE / CREDIT	6	1132.60+	33.55 %
1 - CURRENT BALANCE DUE	1	411.83+	12.20 %
2 - 30-60 DAYS OVERDUE	1	25.00-	-0.74 %
3 - 60-90 DAYS OVERDUE	1	316.79+	9.38 %
4 - OVER 90 DAYS IN ARREARS	1	1539.33+	45.60 %
5 - CREDIT SUSPENDED	0	0.00+	0.00 %
6 - LEGAL ACTION PENDING	0	0.00+	0.00 %
7 - INACTIVE ACCOUNT	0	0.00+	0.00 %
	-----	-----	-----
TOTALS	10	3375.55+	100.00 %

07/17/79 - REPORT # 2

---

## Records With balance Over The Credit Limit

07/17/79 - RECORDS WITH BALANCE OVER THE CREDIT LIMIT - RUN# 1

&lt;ACCT# SEQUENCE&gt;

ACCT#	NAME OF ACCOUNT	ADDRESS	CITY, STATE	ZIP	
PHONE	CRT LMT	2RQ	FC?	DT LAST SALE	DT LAST PYMT
PRV BAL	TTL SLE	PMT RCV	CUR BAL	STATUS	
1006	BILLIE FAIRLEY	14 RIVERVIEW RD	TARRYTOWN, N. Y.	10591	
914-769-7246	1000	10	N	07/21/79	00/00/00
0.00+	1539.33+	0.00+	1539.33+	4 - OVER 90 DAYS IN ARREARS	

RECEIVABLES FOR THESE RECORDS: 1539.33+  
 THIS IS 45.60 % OF 3375.55+ (TOTAL RECEIVABLES).

END OF REPORT





# TRS-80 ORDER FORM FOR CUSTOM PRINTED STATEMENTS

\* DESIGNED FOR USE ON TRS-80 TRACTOR FEED PRINTER

## STANDARD SPECIFICATIONS

SIZE: 7 x 8½ (6 x 8½ DETACHED)

NUMBER OF PARTS: ONE

INK COLORS: BLUE & BLACK OVERPRINT

PAPER: 24LB. WHITE BOND

TO ORDER YOUR FORMS SIMPLY FOLLOW  
THE STEP BY STEP PROCEDURES AS SHOWN  
BELOW: REMEMBER THAT YOUR ACCURACY  
IS VERY IMPORTANT AS YOU WILL BE  
RESPONSIBLE FOR YOUR ERRORS!

#1- IF YOU DESIRE YOU MAY ORDER OUR STOCK FORM WITHOUT YOUR NAME, ADDRESS, OR TERMS PRINTED. CHECK QUANTITY DESIRED. THERE ARE 2,200 FORMS PER CARTON.

☐ 1 CARTON \$ 90.00 ☐ 2 CARTONS \$ 150.00 ☐ 3 CARTONS \$ 195.00 ☐ 4 CARTONS \$ 220.00  
TYPE OR PRINT CLEARLY

#2- IF YOU PREFER PERSONALIZED FORMS PRINT EXACTLY IN THE SPACE AT RIGHT, THE IMPRINT YOU WANT ON YOUR FORM. SPACE SHOWN IS THE EXACT SIZE AVAILABLE FOR YOUR COMPANY NAME, ADDRESS, STATE, ZIP AND PHONE NUMBER AS DESIRED.

#3- IF YOU DESIRE, WE CAN PRE-PRINT YOUR TERMS AT THE BOTTOM OF THE FORM. SPACE SHOWN AT THE RIGHT IS THE EXACT SIZE AVAILABLE. PLEASE TYPE OR PRINT CLEARLY.

CHECK QUANTITY DESIRED FOR PERSONALIZED FORMS. WE SUGGEST YOU ORDER AT LEAST A 1 YEAR SUPPLY.

#4- ☐ 1 CARTON ☐ 2 CARTONS ☐ 3 CARTONS ☐ 4 CARTONS ☐ OTHER QTY. \_\_\_\_\_  
\$ 135.00 \$ 210.00 \$ 255.00 \$ 280.00 Price will be quoted (over 8,800 only)

#5- IF YOU WANT PERSONALIZED #6¼ WINDOW ENVELOPES, PLEASE CHECK QTY. DESIRED. PRINTING WILL BE THE SAME AS SHOWN IN #2 ABOVE, UNLESS YOU GIVE OTHER INSTRUCTIONS.

☐ NO ENVELOPES ☐ 500 \$ 24.16 ☐ 1,000 \$ 37.51 ☐ 2,500 \$ 72.18 ☐ 5,000 \$ 124.56

#6- ☐ CHECK HERE IF YOU HAVE SPECIAL INSTRUCTIONS ON REVERSE.

#7- ENCLOSE YOUR CHECK FOR FULL PAYMENT MADE PAYABLE TO TRINITY FORMS COMPANY. TEXAS RESIDENTS ADD 5% SALES TAX.

#8- MAKE SURE ALL INSTRUCTIONS ARE COMPLETE AND CHECK FOR SPELLING ERRORS.

#9- ALLOW 3 TO 5 WEEKS FOR COMPLETE DELIVERY, 2 WEEKS IF YOU ARE ONLY ORDERING STOCK FORMS.

#10- GIVE US YOUR SHIPPING ADDRESS: \_\_\_\_\_

#11- SIGNED: \_\_\_\_\_ TITLE: \_\_\_\_\_ PHONE #: \_\_\_\_\_

#12- SEND THIS FORM AND YOUR PAYMENT TO:  
TRINITY FORMS COMPANY, NO. 1 FORMS DRIVE, CARROLLTON, TEXAS 75006  
PRICES SUBJECT TO CHANGE WITHOUT NOTICE-



```
10 REM ** ACCOUNTS RECEIVABLE SETUP PROGRAM **
20 REM COPYRIGHT (C) 1979, TANDY CORP.
30 REM VERSION 1.0 - 07/16/79
40 CLS: CLEAR1000: DEFINT A-Z: W$="": Q=0: Q1=0: Q2=0: DIM G(24), G#(24): DIM W, IN$, CF, WD, W$,
    WS, FL, WL: ON ERROR GOTO 600
50 PRINT"                TRS-80 ACCOUNTS RECEIVABLE SYSTEM
                        SETUP PROGRAM"
```

## (I) N I T I A L S E T U P

## (M) O D I F I C A T I O N

```
    ENTER SELECTION --> ";
60 FL=1: GOSUB 280: Q$=IN$: IFCF<>0 THEN 60 ELSE IF Q$<>"I" AND Q$<>"M" THEN PRINT CHR$(8);: GO
    TO 60 ELSE IF Q$="I" THEN ZX=1: GOTO 80 ELSE ZX=0
70 GOSUB 500
80 CLS: IF ZX=0 THEN PRINT@976, "PRESS <ENTER> IF OK AS IS";: PRINT@0,;
90 PRINT"YOUR COMPANY NAME -----> ";: FL=25: GOSUB 280: IFCF<>0 THEN 80 ELSE IF IN$<>"T
    HENN$=IN$
100 PRINT"
    YOUR STREET ADDRESS -----> ";: FL=25: GOSUB 280: IFCF<>0 THEN 80 ELSE IF IN$<>" " THEN A
    $=IN$
110 PRINT"
    CITY, STATE & ZIP CODE -> ";: FL=25: GOSUB 280: IFCF<>0 THEN 80 ELSE IF IN$<>" " THEN C
    SZ$=IN$
120 PRINT"
    YOUR PHONE NUMBER -----> ";: FL=15: GOSUB 280: IFCF<>0 THEN 80 ELSE IF IN$<>" " THEN P
    N$=IN$
130 PRINT"

    ENTER MONTHLY FINANCE CHARGE RATE
    (PER CENT OF UNPAID BALANCE) -----> ";: FL=-4: GOSUB 280: IFCF<>0 THEN 80 ELSE I
    F IN$<>" " THEN FC!=VAL(IN$)/100: PRINT" %" ELSE PRINT
140 PRINT"
    ENTER MINIMUM PAYMENT -----> ";: FL=-10: GOSUB 280: IFCF<>0 THEN 80 ELSE
    IF IN$<>" " THEN MB#=VAL(IN$)
150 PRINT"

    ARE YOU USING BILLING STATEMENT FORMS WITH YOUR
    COMPANY NAME PRE-PRINTED ON THE FORM ? (Y/N) ";
160 FL=1: GOSUB 280: IFCF<>0 THEN 80 ELSE IF IN$=" " THEN 170 ELSE IF IN$<>"Y" AND IN$<>"N" THEN P
    RINT CHR$(8);: GOTO 160 ELSE IF IN$="Y" THEN PP=0 ELSE PP=1
170 IF ZX=0 THEN GOSUB 560: GOSUB 960: IF Q$="I" THEN GOSUB 810: GOTO 190 ELSE IF UT=0 AND PD<>2 TH
    EN GOSUB 810: GOTO 210 ELSE GOTO 210 ELSE 180
180 GOSUB 650: GOSUB 810
190 CLS: PRINT@448, "ENTER YOUR PASSWORD --> ";
200 FL=8: GOSUB 280: IFCF<>0 THEN 190 ELSE IF IN$=" " THEN PRINT STRING$(8,8);: GOTO 200 ELSE P$
```

```

=====
      =IN$
210 CLS:PRINT@448,"ARE ALL ENTRIES CORRECT ? (Y/N) ";
220 FL=1:GOSUB280:R$=IN$:IFCF<>0THEN220ELSEIFR$<>"Y"ANDR$<>"N"THENPRINTCHR$(8);:
      GOTO220ELSEIFR$="N"THENGOSUB560:IFQ$="I"THENZX=0:GOTO80ELSE80ELSE230
230 IFQ$="M"THEN260
240 CLS:PRINT@471,"STORING SETUP DATA":GOSUB470:GOSUB480
250 CLS:PRINT@474,"SETUP STORED

      TYPE:  RUN";CHR$(34);"ARS";CHR$(34);:CLEAR50:END
260 CLS:PRINT@470,"STORING ALL CHANGES":GOSUB490
270 CLS:PRINT@471,"ALL CHANGES STORED

      TYPE:  RUN";CHR$(34);"ARS";CHR$(34);:CLEAR50:END
280 W=FRE(W$):IN$="":CF=0:W$=INKEY$:WD=0:WS=WD:WL=WD:IFFL=WDTHENFL=1
290 PRINTSTRING$(ABS(FL),CHR$(136));STRING$(ABS(FL),CHR$(24));
300 PRINTCHR$(14);:FORW=1TO25:W$=INKEY$:IFW$<>" "THEN310ELSENEXT:PRINTCHR$(15);:F
      ORW=1TO25:W$=INKEY$:IFW$<>" "THEN310ELSENEXT:GOTO300
310 IFW$<>CHR$(13)THEN330ELSEPRINTSTRING$(ABS(FL)-WL," ");
320 PRINTCHR$(15);:W=25:NEXT:RETURN
330 IFW$<>"a"THEN350
340 CF=1:PRINTCHR$(15);:RETURN
350 PRINTCHR$(14);:IFW$=CHR$(24)THENPRINTSTRING$(WL,CHR$(24));:GOTO280
360 IFW$<>CHR$(8)THEN400ELSEIFWL=0THEN300ELSEPRINTCHR$(24);:IFFL>0THEN380ELSEIFP
      EEK(16418)=44THEN390
370 IFPEEK(16418)=46THENWD=0:GOTO380ELSEIFPEEK(16418)=43ORPEEK(16418)=45THENWS=0
380 IN$=LEFT$(IN$,LEN(IN$)-1)
390 WL=WL-1:POKE16418,136:GOTO300
400 IFABS(FL)=WLTHEN300ELSEIFFL>0THENIFW$>=" "ANDW$<="z"THEN450
410 IFW$="."ANDWD=0THENWD=1:GOTO450
420 IFW$=","THENPRINTW$;:WL=WL+1:GOTO460
430 IF(W$="-"ORW$="+")ANDWS=0ANDWL=0THENWS=1:GOTO450
440 IFW$<"0"ORW$>"9"THEN300
450 PRINTW$;:IN$=IN$+W$:WL=WL+1
460 IFABS(FL)=1THEN320ELSE300
470 PI$="CUSINDEX."+P$+"":1":PD$="CUSDATA."+P$+"":1":PT$="TRANSACTION."+P$+:PS$="CUSSE
      TUP."+P$+"":1":PG$="GLFILE."+P$+"":1":IFPD=2THENPT$=PT$+"":2":RETURNELSEPT$=PT
      $+"":1":RETURN
480 OPEN"O",1,PI$:TI=0:TN=1:F=0:U=0:UT=0:UN=1:R0=1:R1=1:R2=1:R3=1:R4=1:R5=1:R6=1
      :R7=1:CV#=0:EP=0:PRINT#1,TI;TN;F;U;UT;UN;EP;R0;R1;R2;R3;R4;R5;R6;R7;STR$(CV
      #)+"D0":FORN=1TOTN-1:PRINT#1,P(N,0);P(N,1);P(N,2):NEXT:CLOSE1
490 OPEN"R",1,PD$:CLOSE1:OPEN"R",1,PT$:CLOSE1:OPEN"O",1,PS$:PRINT#1,N$:PRINT#1,A
      $:PRINT#1,CSZ$:PRINT#1,PN$:PRINT#1,FC!:PRINT#1,GT;PD;PC;PT;PP:PRINT#1,GL$:P
      RINT#1,MB#:CLOSE1:IFRF=0THENGOSUB770:RETURNELSERETURN
500 CLS:PRINT@462,"ENTER PASSWORD --> ";:P$="":FORX=1TO8
510 PE$=INKEY$:IFPE$=" "THEN510ELSEIFPE$<" "ORPE$>"Z"THENIFASC(PE$)=13THENGOSUB47
      0:GOTO530ELSE510
520 PRINT"#";:P$=P$+PE$:NEXT:GOSUB470
530 OPEN"I",1,PS$:LINEINPUT#1,N$:LINEINPUT#1,A$:LINEINPUT#1,CSZ$:LINEINPUT#1,PN$
      :INPUT#1,FC!,GT,PD,PC,PT,PP:LINEINPUT#1,GL$:INPUT#1,MB#:CLOSE1:OPEN"I",1,PI
      $:INPUT#1,TI,TN,F,U,UT,UN:CLOSE1

```

```
=====
540 OPEN"R",3,PG#:RF=0:FORG=1TOGT:GOSUB900:GOSUB940:GOSUB950:G(G)=GA:G#(G)=GC#:I
    FGC#<>0THENRF=1
550 NEXT:CLOSE3
560 CLS:PRINTTAB(21)"CURRENT COMPANY DATA":PRINT"
    YOUR COMPANY NAME:";TAB(30)N$;"
    YOUR ADDRESS:";TAB(30)A$;"
    CITY, STATE & ZIP CODE:";TAB(30)CSZ$;"
    PHONE NUMBER:";TAB(30)PN$
570 PRINT"

    MONTHLY FINANCE CHARGE RATE: ";TAB(36);USING"##.##";FC!*100;PRINT" %"
580 PRINT"MINIMUM PAYMENT CUT-OFF POINT: ";USING"#####.##";MB#;PRINT"

    COMPANY NAME IS";IFPP=1THENPRINT" NOT";
590 PRINT" PRE-PRINTED ON THE BILLING STATEMENT FORMS.":PRINT"
    MAX CAPACITY: ";PC;"CUSTOMER ACCOUNTS / ";PT;"TRANSACTIONS/MONTH":GOSUB630:
    RETURN
600 IFERR/2+1=70THENCLS:PRINT@465,"*** INVALID PASSWORD ***

    ";TAB(20)"PROGRAM TERMINATED
    ":CLEAR50:END
610 IFERR/2+1=54THENCLS:PRINT@465,"NO FILES ON DATA DISK":PRINT:PRINT:CLEAR50:EN
    D
620 ONERRORGOTO00
630 IN#=INKEY$:PRINT@976,"PRESS <ENTER> TO CONTINUE ";
640 EN#=INKEY$:IFEN#<>CHR$(13)THEN640ELSEReturn
650 CLS:PRINT"          ASSIGNMENT OF GENERAL LEDGER CODES (GLC)
    ":FORG=1TO24:G#(G)=0:NEXT:PRINT"GLC: 1&2  ACCOUNTS RECEIVABLE (A/R)
    GL ACCOUNT # : ";:FL=-4:GOSUB280:IFCF<>0THEN650ELSEG(1)=VAL(IN$):G(2)=G(1)
660 PRINT"
    GLC: 3      ASSESSED FINANCE CHARGES (F/C)      GL ACCOUNT # : ";:FL=-4:GOSUB28
    0:IFCF<>0THEN650ELSEG(3)=VAL(IN$)
670 PRINT"
    GLC: 4      BAD DEBTS WRITE-OFF (B/D)          GL ACCOUNT # : ";:FL=-4:GOSUB28
    0:IFCF<>0THEN650ELSEG(4)=VAL(IN$)
680 PRINT"

    ENTER UP TO 20 GLC'S TO BE USED FOR POSTING TO THE GENERAL
    LEDGER.  AFTER THE LAST GLC, TYPE -1 TO END THIS SECTION.
    ":GL$="A/R  A/R  F/C  B/D  "
690 FORG=5TO24:PRINTUSING"%      % ###      %      % ####";"GLC : " ;G;"GL ACC
    OUNT # : ";
700 FL=-4:GOSUB280:IFCF<>0THEN650
710 IFVAL(IN$)<-1THENG(G)=VAL(IN$)ELSEPRINT:G=G-1:GOTO760
720 PRINTTAB(39)"NAME ABBREVIATION: ";
730 FL=5:GOSUB280:IFCF=0THENIN$=IN$+"      ":GL$=GL$+LEFT$(IN$,5)ELSE650
740 PRINT
750 NEXT:G=G-1
760 GT=G:GOSUB1170:RETURN
770 OPEN"R",3,PG#:FORG=1TOGT
```

```

=====
780 GA=G(G):GC#=G#(G)
790 GOSUB900:GOSUB920
800 NEXT:CLOSE3:RETURN
810 CLS:PRINT"ARS STORAGE CAPACITY OPTIONS

"
820 PRINT"1 - 300 CUSTOMER ACCOUNTS / 1000 TRANSACTIONS/MONTH
      (2 DISK SYSTEMS)

      2 - 100 CUSTOMER ACCOUNTS / 2000 TRANSACTIONS/MONTH
      (2 DISK SYSTEMS)

      3 - 500 CUSTOMER ACCOUNTS / 2500 TRANSACTIONS/MONTH
      (3 DISK SYSTEMS)

830 PRINT"SELECT 1, 2 OR 3 ---> ";
840 FL=-1:GOSUB280:IFCF<>0THEN840ELSEIFIN$<"1"ORIN$>"3"THENPRINTCHR$(8):GOTO840
850 PRINT:IFPC=300ANDIN$="2"ANDQ$="M"THEN810ELSEONVAL(IN$)GOTO860,870,880
860 PD=1:PC=300:PT=1000:GOTO890
870 PD=1:PC=100:PT=2000:GOTO890
880 PD=2:PC=500:PT=2500:IFQ$="M"THENKILLPT$:PT#=LEFT$(PT$,LEN(PT$)-1)+"2":OPEN"R
      ",3,PT$:CLOSE3
890 RETURN
900 GR=INT((G-1)/25)+1:GD=G-25*INT((G-1)/25)-1
910 FIELD3,GD*10ASG0$,2ASG1$,8ASG3$:RETURN
920 LSETG1$=MKI$(GA):LSETG3$=MKD$(GC#)
930 PUT3,GR:RETURN
940 GET3,GR:RETURN
950 GA=CVI(G1$):GC#=CVD(G3$):RETURN
960 IFRF=0THEN990
970 CLS:PRINT@448,"GLC FILE CONTAINS DATA."
980 PRINT"REVISION NOT ALLOWED PRIOR TO END OF PERIOD PROCESSING.":PRINT:GOSUB63
      0:RETURN
990 CLS:PRINT"PRESS <ENTER> IF OK AS IS.
      TYPE -1 AFTER LAST VALID CODE.":PRINT
1000 GL$=LEFT$(GL$+STRING$(120,32),120)
1010 PRINT"GLC : 1&2   ";G(1);"A/R";TAB(30)"GL ACCT # : ";
1020 FL=-4:GOSUB280:IFCF<>0THEN960ELSEIFIN$<>" "THENG(1)=VAL(IN$):G(2)=G(1):PRINT
      ELSEPRINT
1030 PRINT"GLC : 3     ";G(3);"F/C";TAB(30)"GL ACCT # : ";
1040 FL=-4:GOSUB280:IFCF<>0THEN960ELSEIFIN$<>" "THENG(3)=VAL(IN$):PRINTELSEPRINT
1050 PRINT"GLC : 4     ";G(4);"B/D";TAB(30)"GL ACCT # : ";
1060 FL=-4:GOSUB280:IFCF<>0THEN960ELSEIFIN$<>" "THENG(4)=VAL(IN$):PRINTELSEPRINT
1070 FORG=5TO24
1080 PRINTUSING"%    % ##      #### %    %";"GLC :";G(G);MID$(GL$, (G-1)*5+1,5);:P
      RINTTAB(30)"GL ACCT # : ";
1090 FL=-4:GOSUB280:IFCF<>0THEN960
1100 IFVAL(IN$)=-1THENPRINT:G=G-1:GOTO1160

```

```

1110 IFIN$<>" " THEN G(G)=VAL(IN$)
1120 PRINT TAB(50)"NAME : ";
1130 FL=5:GOSUB 280:IFCF<>0 THEN 960
1140 IFIN$<>" " THEN IN$=LEFT$(IN$+"      ",5):MID$(GL$, (G-1)*5+1,5)=IN$:PRINT ELSE PR
    INT
1150 NEXT G=G-1
1160 GT=G:GL$=LEFT$(GL$,GT*5):GOSUB 1170:RETURN
1170 PRINT@960,"WOULD YOU LIKE A PRINTOUT OF THE GLC'S ? (Y/N) ";
1180 FL=1:GOSUB 280:IFCF<>0 THEN 1180 ELSE IFIN$="N" THEN RETURN ELSE IFIN$="Y" THEN 1190 EL
    SE PRINT CHR$(8):GOTO 1180
1190 IF PEEK(14312)>127 THEN PRINT@960,CHR$(30);"P R I N T E R   N O T   R E A D Y
    ! !";FOR ZQ=1 TO 1500:NEXT:GOTO 1170 ELSE LPRINT"GLC      DESCRIPCION      ACCOUNT
    #";TAB(57)"NOTES":LPRINT STRING$(80,"-")
1200 FOR G=1 TO GT:LPRINT USING"###          %      %          ####";G:MID$(GL$, (G-1)*5+1
    ,5):G(G):LPRINT STRING$(80,"-"):NEXT:LPRINT CHR$(12):RETURN

```

```

10 REM ** ACCOUNTS RECEIVABLE PROGRAM **
20 REM COPYRIGHT (C) 1979, TANDY CORP.
30 REM VERSION 1.0 - 07/16/79
40 CLEAR600:DEFINTA-Z:DIMP(500,2):W$="":N#=0:W#=0:V$="":X=0:D1#=16777216:D2#=655
   36:D3#=256
50 DIM W,IN$,CF,WD,W$,WS,FL,WL
60 ONERRORGOTO110:OPEN"I",1,"TRANSFER"
70 LINEINPUT#1,PI$:LINEINPUT#1,PD$
80 LINEINPUT#1,PS$:LINEINPUT#1,PT$:LINEINPUT#1,PG$
90 CLOSE1:KILL"TRANSFER"
100 GOTO220
110 IFERR/2+1=54THENRESUME130
120 ONERRORGOTO0
130 IFLEFT$(TIME$,2)=""00"ORMID$(TIME$,3,1)<>"/"ORMID$(TIME$,6,1)<>"/"THENCLS:GOT
   0160
140 CLS:PRINT@458,"DATE IS SET TO ";LEFT$(TIME$,8);" . IS THIS CORRECT ? (Y/N) "
   ;
150 FL=1:GOSUB390:IFIN$<>"Y"ANDIN$<>"N"THENPRINTCHR$(8);:GOTO150ELSEIFIN$="Y"THE
   N210
160 PRINT@588,"ENTER CORRECT DATE (MM/DD/YY) : ";
170 FL=8:GOSUB390:DT$=IN$
180 POKE16454,VAL(LEFT$(DT$,2)):POKE16452,VAL(RIGHT$(DT$,2))
190 POKE16453,VAL(MID$(DT$,4,2))
200 GOTO130
210 CLS:GOSUB3220:GOSUB380:PRINT@472,"INITIALIZING"
220 ONERRORGOTO3540
230 POKE16424,67:POKE16425,1
240 READL0$,L1$,L2$,L3$,T0$,T1$,T2$,T3$,T4$,T5$,T6$,T7$,T8$,T9$,TA$,TB$,TX$,TD$,
   TE$,TF$,TG$,TH$,TI$,TJ$,TK$,TL$,TM$,TN$,SS$:SV$="":FORX=1TO36:READSV:SV$=SV
   $+CHR$(SV):NEXT:RESTORE
250 XX=1:LF$=" ":LL$=STRING$(80,"-")
260 GOSUB3350:J=1:GOSUB2480
270 GOSUB380:PRINT"FUNCTIONS:

   *( A ) D D   A C C O U N T S
   *( C ) U S T O M E R   A C C O U N T S
   *( T ) R A N S A C T I O N S
   *( P ) O S T   T R A N S A C T I O N S
   *( R ) E P O R T S
   *( E ) N D   O F   P E R I O D
   *( S ) Y S T E M   S T A T U S"
280 PRINT"*( @ ) E X I T   P R O G R A M "
290 PRINT:PRINT"SELECT FUNCTION --> ";
300 FL=1:GOSUB390:IFCF=1THEN370ELSE$=INSTR("ACTPRES",IN$):IFS=0THENPRINTCHR$(8);
   :GOTO300
310 IFS>1AND$<7ANDTI=0THENCLS:PRINT@470,"NO ACCOUNTS ON FILE":FOR@0=1TO1500:NEXT
   :GOTO270
320 ONSGOTO610,1080,1580,1630,2470,2460,3460
330 IFFD=0THEN270ELSECLS:PRINT@474,"SORTING DATA":GOSUB340:CLEAR50:RUN"ARSORT":E
   ND

```

```
-----
340 CLOSE:GOSUB3430:OPEN"O",1,"TRANSFER:1"
350 PRINT#1,PI$:PRINT#1,PD$:PRINT#1,PS$:PRINT#1,PT$:PRINT#1,PG$
360 CLOSE1:RETURN
370 CLS:PRINT@473,"END OF PROGRAM

      ":GOSUB3430:CLEAR50:END
380 CLS:PRINTTAB(15)"TRS-80 ACCOUNTS RECEIVABLE SYSTEM":PRINT@117,LEFT$(TIME$,8)
      :RETURN
390 WD=FRE("")
400 IN$="":CF=0:W$=INKEY$:WD=0:WS=WD:WL=WD:IFFL=WDTHENFL=1
410 PRINTSTRING$(ABS(FL),CHR$(136));STRING$(ABS(FL),CHR$(24));
420 PRINTCHR$(14);:FORW=1TO25:W$=INKEY$:IFW$<>" "THEN430ELSENEXT:PRINTCHR$(15);:F
      ORW=1TO25:W$=INKEY$:IFW$<>" "THEN430ELSENEXT:GOTO420
430 IFW$<>CHR$(13)THEN450ELSEPRINTSTRING$(ABS(FL)-WL," ");
440 PRINTCHR$(15);:W=25:NEXT:RETURN
450 IFW$<>"@"THEN470
460 CF=1:PRINTCHR$(15);:RETURN
470 PRINTCHR$(14);:IFW$=CHR$(24)THENPRINTSTRING$(WL,CHR$(24));:GOTO400
480 IFW$<>CHR$(8)THEN520ELSEIFWL=0THEN420ELSEPRINTCHR$(24);:IFFL>0THEN500ELSEIFP
      EEK(16418)=44THEN510
490 IFPEEK(16418)=46THENWD=0:GOTO500ELSEIFPEEK(16418)=43ORPEEK(16418)=45THENWS=0
500 IN$=LEFT$(IN$,LEN(IN$)-1)
510 WL=WL-1:POKE16418,136:GOTO420
520 IFABS(FL)=WLTHEN420ELSEIFFL>0THENIFW$>=" "ANDW$<="z"THEN570
530 IFW$="."ANDWD=0THENWD=1:GOTO570
540 IFW$=","THENPRINTW$;:WL=WL+1:GOTO580
550 IF(W$="-"ORW$="+")ANDWS=0ANDWL=0THENWS=1:GOTO570
560 IFW$<"0"ORW$>"9"THEN420
570 PRINTW$;:IN$=IN$+W$:WL=WL+1
580 IFABS(FL)=1THEN440ELSE420
590 PRINT@128,CHR$(31);"(TYPE @ TO EXIT TO MENU)":RETURN
600 PRINT@128,CHR$(30);"(TYPE @ TO REDO ACCOUNT)":RETURN
610 J=TN
620 CLS:PRINTTAB(26)"ADD ACCOUNTS
      ":PRINT"
      FUNCTIONS:

      >(O)LD  ACCOUNT
      >(N)EW  ACCOUNT
      >(a)EXIT TO  MENU

      SELECT FUNCTION --> ":FL=1:GOSUB390:IFCF=1THENGOTO330ELSEZ9$=IN$
630 CLS:IFZ9$="O"THENPRINTTAB(20)"ENTER OLD ACCOUNT"ELSEIFZ9$="N"THENPRINTTAB(20)
      )"ENTER NEW ACCOUNT"ELSE620
640 IFTI=PCTHENCLS:PRINT@468,"SYSTEM IS FULL (":PC;"ACCOUNTS)":GOSUB3320:GOTO330
650 GOSUB590:PRINT@256,CHR$(30);:GOSUB2630:GOSUB660:GOTO670
660 IFFD<>0THENPRINT@960,"( LAST ACCT.# :":IL;"/ NAME : ":NL$;" / # OF ACCTS":TI
```

```
;"":RETURNELSERETURN
670 PRINT@280,;:FL=-4:GOSUB390:IFCF=0THENID=VAL(IN$)ELSE620
680 IFIN$<>" "THEN690ELSE670
690 GOSUB600
700 PRINT@344,;:FL=11:GOSUB390:IFCF=0THENF$=IN$+STRING$(11-LEN(IN$),0)ELSE630
710 PRINT@408,;:FL=13:GOSUB390:IFCF=0THENL$=IN$ELSE630
720 IFL$=" "THEN710
730 PRINT@472,;:FL=17:GOSUB390:IFCF=0THENS$=IN$ELSE630
740 PRINT@536,;:FL=15:GOSUB390:IFCF=0THENC$=IN$ELSE630
750 PRINT@600,;:FL=5:GOSUB390:IFCF=0THENZ$=IN$ELSE630
760 IFZ$=" "THEN750
770 PRINT@664,;:FL=12:GOSUB390:IFCF=0THENPH$=IN$ELSE630
780 PRINT@728,;:FL=-6:GOSUB390:IFCF=0THENC$=VAL(IN$)ELSE630
790 PRINT@792,;:FL=-3:GOSUB390:IFCF=0THENR$=VAL(IN$)ELSE630
800 IFR$>100THEN790ELSEPRINT"%";
810 PRINT@856,;:FL=1:GOSUB390:IFCF=0AND(IN$="Y"ORIN$="N")THENFC$=IN$ELSEIFCF=1TH
    EN630ELSE810
820 IFZ$="N"THENPB#=0:TS#=0:PR#=0:CB#=0:DS!=0:DP!=0:ST=0:GOTO920
830 CLS:PRINTTAB(22)"OLD ACCOUNT - PAGE 2":PRINTTAB(20)"(TYPE @ TO REDO ACCOUNT)

    ":GOSUB2640:GOSUB660
840 PRINT@216,;:FL=-10:GOSUB390:GOSUB950:IFCF=0THENPB#=VAL(IN$):PRINT@216,CHR$(3
    0);USINGL2$;PB#;ELSEONCFGOTO630,840
850 PRINT@280,;:FL=-10:GOSUB390:GOSUB950:IFCF=0THENT$=VAL(IN$):PRINT@280,CHR$(3
    0);USINGL2$;T$;ELSEONCFGOTO630,850
860 PRINT@344,;:FL=8:GOSUB390:GOSUB980:IFCF=0THENDS!=DX!:PRINTCHR$(30);ELSEONCFG
    OTO630,860
870 PRINT@408,;:FL=-10:GOSUB390:GOSUB950:IFCF=0THENPR#=-VAL(IN$):PRINT@408,CHR$(
    30);USINGL2$;PR#;ELSEONCFGOTO630,870
880 PRINT@472,;:FL=8:GOSUB390:GOSUB980:IFCF=0THENDP!=DX!:PRINTCHR$(30);ELSEONCFG
    OTO630,880
890 CB#=PB#+PR#+TS#:PRINT@536,USINGL2$;CB#
900 GOSUB2680
910 PRINT@621,;:FL=-1:GOSUB390:IFCF<>0THEN630ELSEST=VAL(IN$):IFST<0ORST>7THENPRI
    NTCHR$(8);:GOTO910
920 TC=0:CV#=CV#+CB#:TI=TI+1:FD=1:DI=0:DJ=0
930 GOSUB2740:GOSUB2480:GOSUB2560:GOSUB2500:TN=TN+1:J=TN
940 IL=ID:N$=L$:GOTO630
950 IFCF<>0THENRETURN
960 IFLEN(IN$)>=3THENIFMID$(IN$,LEN(IN$)-2,1)=". "THENRETURN
970 GOSUB1030:RETURN
980 IFIN$=" "THENDX!=0:RETURN
990 IFLEN(IN$)<>8ORMID$(IN$,3,1)<>"/"ORMID$(IN$,6,1)<>"/"THENGOSUB1030:RETURN
1000 DX$=MID$(IN$,7,2)+MID$(IN$,1,2)+MID$(IN$,4,2)
1010 FORW=1TO6:IFMID$(DX$,W,1)<"0"ORMID$(DX$,W,1)>"9"THENW=6:NEXT:GOSUB1030:RETU
    RNELSENEXT
1020 DX!=VAL(DX$):RETURN
1030 PRINT@960,CHR$(30);:FORW9=1TO4:FORW8=1TO15:PRINT@980,"INVALID FORMAT !";:NE
    XTW8:FORW8=1TO15:PRINT@980,CHR$(30);:NEXTW8:NEXTW9:CF=2:GOSUB660:RETURN
1040 DX$=STR$(DX!):DX$=RIGHT$(DX$,LEN(DX$)-1)
```

```
1050 IFLEN(DX$)<6THENFORW=1TO6-LEN(DX$):DX$="0"+DX$:NEXT
1060 IN$=MID$(DX$,3,2)+"/"+MID$(DX$,5,2)+"/"+MID$(DX$,1,2)
1070 RETURN
1080 CLS:PRINTTAB(20)"CUSTOMER ACCOUNTS"
1090 PRINT@128,CHR$(31);:GOSUB2630
1100 PRINT@971,"(TYPE @ TO EXIT TO MAIN MENU)";
1110 PRINT@896,"ACCESS BY: (A)CCOUNT NUMBER OR BY (N)AME ? ";
1120 FL=1:GOSUB390:IFCF<>0THENGOTO330ELSEIFIN$<>"A"ANDIN$<>"N"THENPRINTCHR$(8);:
    GOTO1120
1130 PRINT@960,CHR$(30);:IFIN$="A"THEN AM=0 ELSE AM=1
1140 ON AM+1GOTO1150,1170
1150 GOSUB3190:PRINT@151;:
1160 FL=-4:GOSUB390:IFCF=0THENL$=IN$:GOTO1190ELSEGOTO1080
1170 GOSUB3190:PRINT@280;:
1180 FL=13:GOSUB390:IFCF=0THENL$=IN$:GOTO1190ELSEGOTO1080
1190 GOSUB2820:Z8=N:IFKS=0THENPRINT@896,CHR$(30);TAB(20)"** ACCOUNT NOT FOUND **
    ";:GOSUB3340:GOTO1080
1200 GOSUB3140
1210 GOSUB3170
1220 FL=1:GOSUB390:IFCF<>0THENGOTO330
1230 O$=IN$:IFO$<>"U"ANDO$<>"S"ANDO$<>"N"ANDO$<>"P"ANDO$<>"D"ANDO$<>"T"ANDO$<>"H
    "THENPRINTCHR$(8);:GOTO1220
1240 IFO$="T"THENONPTGOSUB3140,3150:GOTO1210
1250 IFO$="H"THENGOSUB3140:GOSUB3900:GOSUB3150:FORX=1TO1000:NEXT:GOSUB3900:GOSUB
    3140:GOTO1210
1260 IFO$="S"THEN1080
1270 IFO$="P"THENNZ=N:FP=-1:GOSUB1280:GOTO1210ELSEIFO$="N"THENNZ=N:FP=1:GOSUB128
    0:GOTO1210ELSEGOTO1290
1280 N=N+FP:IFN>TN-1THENPRINT@985,"LAST ACCOUNT";:GOSUB3340:N=NZ:J=P(N,AM):RETUR
    NELSEIFN<1THENPRINT@985,"FIRST ACCOUNT";:GOSUB3340:N=NZ:J=P(N,AM):RETURN
1285 J=P(N,AM):IFJ<0THEN1280ELSEGOSUB2480:GOSUB2560:GOSUB2570:GOSUB3140:RETURN
1290 IFO$<>"D"THEN1390ELSEIFP(N,AM)<0THEN1210
1300 IFDI<>0ANDU<>0THENPRINT@832,CHR$(31);"CANNOT DELETE THIS ACCOUNT UNTIL AFTE
    R POSTING":GOSUB3340:GOTO1210ELSEIFPT=1THENGOSUB3140
1310 PRINT@768,CHR$(31);STRING$(64,95);"OK TO DELETE ? (Y/N) ";
1320 FL=1:GOSUB390:IFCF<>0THEN1210ELSEIFIN$<>"Y"ANDIN$<>"N"THENPRINTCHR$(8);:GOT
    O1320ELSEIFIN$="N"THEN1210
1330 PRINT@832,CHR$(31);"DELETION IN PROGRESS"
1340 FOR ND=1TOTN-1
1350 FOR NE=0TO2
1360 IFP(ND,NE)=JTHENP(ND,NE)=-J
1370 NEXT NE,ND:F=F+1:CV#=CV#-CB#:TI=TI-1:K=UN
1380 VA$="D":VB=ID:IN$=LEFT$(TIME$,8):GOSUB980:VC!=DX!:VD!=0:VI$="N":VJ$=CHR$(4)
    :VK$=CHR$(0):VL$=CHR$(0):VE#=-CB#:VF#=0:VG#=0:VH#=VE#:GOSUB1620:GOSUB1590:G
    OSUB1610:CLOSE3:PRINT@157,"(DELETED)";:GOTO1210
1390 IFO$<>"U"THEN1210
1400 PRINT@832,CHR$(31);"ENTER LINE NUMBER TO BE CHANGED: ";
1410 FL=-1:GOSUB390:IFCF<>0THEN1210
1420 UP=VAL(IN$):IFUP<10RUP>8THENPRINTCHR$(8);:GOTO1410
1430 IFUP<8ANDPT=1THENGOSUB3140:PRINT@768,STRING$(64,95);CHR$(31);:ELSEIFUP<8AND
```

```
PT=2THENGOSUB3190
1440 ONUPGOTO1450,1460,1470,1480,1490,1500,1520,1530
1450 PRINTT3$;" ";;FL=17:GOSUB390:IFCF=0THENSA$=IN$:GOTO1560ELSE1210
1460 PRINTT4$;" ";;FL=15:GOSUB390:IFCF=0THENC$=IN$:GOTO1560ELSE1210
1470 PRINTT5$;" ";;FL=5:GOSUB390:IFCF=0THENZ$=IN$:FD=1:GOTO1560ELSE1210
1480 PRINTT6$;" ";;FL=12:GOSUB390:IFCF=0THENPH$=IN$:GOTO1560ELSE1210
1490 PRINTT7$;" ";;FL=-6:GOSUB390:IFCF=0THENCL!=VAL(IN$):GOTO1560ELSE1210
1500 PRINTT8$;" ";;FL=-3:GOSUB390:IFCF=0THENRQ=VAL(IN$)ELSE1210
1510 IFRQ>100THEN1500ELSEPRINT"%";:GOTO1560
1520 PRINTT9$;" ";;FL=1:GOSUB390:IFCF=0AND(IN$="Y"ORIN$="N")THENFC$=IN$:GOTO1560
ELSEIFCF=1THEN1210ELSE1520
1530 IFPT=2THENGOSUB3150:PRINT@768,STRING$(64,95);"SELECT NEW STATUS CODE: ";:EL
SEGOSUB3190:PRINT"SELECT NEW STATUS CODE: ";
1540 FL=-1:GOSUB390:IFCF<>0THEN1210
1550 ST=VAL(IN$):IFST<0ORST>7THENPRINTCHR$(8);:GOTO1540
1560 GOSUB2500:IFUP=8THENGOSUB3150ELSEGOSUB3140
1570 GOTO1210
1580 CLS:IFEP=0THENPRINT@474,"TRANSACTIONS":GOSUB340:CLEAR50:RUN"ENTRAN":END:ELS
EPRINT@462,"END OF PERIOD PROCESSING INCOMPLETE":GOSUB3340:GOTO330
1590 IN$=STR$(VE#):GOSUB3800:VE$=V$:IN$=STR$(VF#):GOSUB3800:VF$=V$
1600 IN$=STR$(VG#):GOSUB3800:VG$=V$:RETURN
1610 GOSUB3010:GOSUB3050:GOSUB3030:U=U+1:UT=UT+1:UN=UN+1:RETURN
1620 CLOSE3:OPEN"R",3,PT$:RETURN
1630 CLS:PRINTTAB(23)"POST TRANSACTIONS":PRINT
1640 IFUN=1THENCLS:PRINT@468,"TRANSACTION FILE IS EMPTY":GOSUB3340:GOTO330
1650 PRINT"OPTIONS:

NOT( L ) I S T   A L L   T R A N S A C T I O N S

NOT( P ) O S T   A L L   T R A N S A C T I O N S

NOT( G ) E N E R A L   L E D G E R   R E C A P

NOT( @ ) E X I T   T O   M A I N   M E N U"
1660 PRINT
1670 PRINT"

SELECT OPTION --> ";
1680 FL=1:GOSUB390:IFCF<>0THENGOTO330ELSET$=INSTR("LPG",IN$):IFTS=0THENPRINTCHR$
(8);:GOTO1680
1690 ONTSGOTO1700,1820,2240
1700 GOSUB1620:GOSUB3200:IFCF<>0THEN1630
1710 LPRINTLEFT$(TIME$,8);" : COMPLETE TRANSACTION FILE":LPRINTLF$
1720 GOSUB2410:L$=""
1730 FORK=1TOUN-1:GOSUB3010:GOSUB3050:GOSUB3060
1740 GOSUB1780
1750 GOSUB3210:IFCF=2THENC$=0:GOSUB2410
1760 GOSUB2440
1770 NEXT:CLOSE3:LPRINTLF$:LPRINT"END OF LISTING":LPRINTCHR$(12):GOTO1630
1780 IFVA$="S"THENVA$="SALE":A1=1 ELSEIFVA$="P"THENVA$="PMNT":A1=2 ELSEIFVA$="B"
```

```
=====
THENVA$="B/D":A1=2 ELSEVA$="B/D":A1=2:RM$="GL FILE":GOTO1800
1790 IFVI$="N" THENRM$="NORMAL" ELSEIFVI$="C" THENRM$="CORRECTION" ELSEIFVI$="A" THEN
    RM$="SUPERSEDED" ELSEIFVI$="B" THENRM$="CORRECTION PENDING"
1800 IFSGN(VB)=1 THENRM$=RM$+"-UNPOSTED" ELSERM$=RM$+"-POSTED"
1810 RETURN
1820 IFU=0 THENCLS:PRINT@468,"NO UNPOSTED TRANSACTIONS":GOSUB3340:GOTO1630 ELSECLO
    SE3:OPEN"R",3,PT$:GOSUB3200:IFCF<>0 THEN1630
1830 FORG=1 TOGT:G#(G)=0:NEXT
1840 LPRINTLEFT$(TIME$,8);" - POSTING REPORT #":R0
1850 GOSUB2410
1860 FORPR=1 TOUN-1:K=PR:GOSUB3010:GOSUB3050
1870 IF(SGN(CVI(V1$))=-1 ANDV7$<>"B") ORV7$="A" THEN1980 ELSEGOSUB3060
1880 IFVA$="D" THENGOSUB2040:VB=-ABS(VB):GOSUB3030:GOSUB1780:L$=" ":GOTO1960
1890 AM=0:L$=STR$(ABS(VB))
1900 GOSUB2820:IFKS=0 THEN1950
1910 IFVI$="B" THEN2000 ELSEIFVA$="P" THEN1930 ELSEIFVA$="B" THENPB#=0D0:CB#=0D0:CV#=
    INT(((CV#+VE#)*100D0)+.5D0)/100D0:TS#=0D0:PR#=0D0:TC=-1:GOTO1940
1920 DS!=VC!:TS#=TS#+VH#:CB#=CR#+VH#:CV#=CV#+VH#:GOTO1940
1930 DP!=VC!:PR#=PR#+VE#:CB#=CB#+VE#:CV#=CV#+VE#
1940 VB=-ABS(VB):TC=TC+1:GOSUB2500:GOSUB3030:GOSUB2040
1950 GOSUB1780:IFKS=0 ANDVA$<>"D" THENRM$="INVALID ID#-NOT POSTED"
1960 GOSUB3210:IFCF=2 THENCF=0:GOSUB2410
1970 GOSUB2440
1980 NEXT:GOSUB2080:R0=R0+1:IFR0>32767 THENR0=1
1990 U=0:LPRINTLF$:LPRINT"END OF REPORT":LPRINTLL$:LPRINTCHR$(12):CLOSE3:GOTO163
    0
2000 IFVA$="P" THEN2020
2010 TS#=TS#-VH#:CB#=CR#-VH#:CV#=CV#-VH#:G#(VJ)=G#(VJ)-VE#:G#(VK)=G#(VK)-VF#:G#(
    VL)=G#(VL)-VG#:G#(1)=G#(1)-VH#:GOTO2030
2020 PR#=PR#-VE#:CB#=CB#+VE#:CV#=CV#+VE#:G#(VJ)=G#(VJ)-VE#:G#(2)=G#(2)-VE#
2030 VI$="A":TC=TC-1:VB=-ABS(VB):GOSUB2500:GOSUB3030:GOTO1950
2040 G#(VJ)=G#(VJ)+VE#:G#(VK)=G#(VK)+VF#:G#(VL)=G#(VL)+VG#
2050 IFVA$="S" THENG#(1)=G#(1)+VH#
2060 IFVA$="P" ORVA$="B" THENG#(2)=G#(2)+VE#
2070 RETURN
2080 LPRINTCHR$(12):LPRINTLF$:LPRINT"GENERAL LEDGER INPUTS":LPRINTLF$
2090 LPRINTTAB(20)"DEBIT(+)":TAB(50)"CREDIT(-)"
2100 LPRINTTAB(20)"-----":TAB(50)"-----"
2110 LPRINTLF$
2120 OPEN"R",1,PG$:G=1:GOSUB3080:GOSUB3120:GOSUB3130:AR=GA:GC#=GC#+G#(1):GOSUB31
    00:G=2:GOSUB3080:GOSUB3120:GOSUB3130:GC#=GC#+G#(2):GOSUB3100
2150 FORG=3 TOGT:GOSUB3080:GOSUB3120:GOSUB3130
2160 AN$=MID$(GL$(G-1)*5+1,5)
2170 LPRINT"GLC:";G:IFSGN(G#(G))=-1 THEN2200
2180 LPRINTTAB(10)AR:"/ A/R":TAB(25);USINGL2$;G#(G);
2190 LPRINTTAB(40)GA:"/ ";AN$:TAB(55);USINGL2$;-G#(G):GOTO2220
2200 LPRINTTAB(10)GA:"/ ";AN$:TAB(25);USINGL2$;-G#(G);
2210 LPRINTTAB(40)AR:"/ A/R":TAB(55);USINGL2$;G#(G)
2220 GC#=GC#+G#(G):GOSUB3100
2230 NEXT:CLOSE1:RETURN
```

```
2240 CLOSE1:OPEN"R",1,PG$:GOSUB3200:IFCF<>0THEN1630
2250 LPRINTLEFT$(TIME$,8);" - GENERAL LEDGER RECAP ";
2260 LPRINT"(TOTALS THIS MONTH TO DATE) - REPORT #";R1
2270 LPRINTLL$:LPRINTLF$
2280 LPRINTTAB(20)"DEBIT(+)" ;TAB(50)"CREDIT(-)"
2290 LPRINTTAB(20)"-----" ;TAB(50)"-----":LPRINTLF$
2300 G=1:GOSUB3080:GOSUB3120:AR=CVI(G1$)
2320 FORG=3TOGT:GOSUB3080:GOSUB3120:GOSUB3130
2330 AN$=MID$(GL$, (G-1)*5+1,5)
2340 LPRINT"GLC:" ;G;:IFSGN(GC#)=-1THEN2370
2350 LPRINTTAB(10)AR;" / A/R" ;TAB(25)USINGL2$;GC#;
2360 LPRINTTAB(40)GA;" / " ;AN$;TAB(55)USINGL2$;-GC#;GOTO2390
2370 LPRINTTAB(10)GA;" / " ;AN$;TAB(25)USINGL2$;-GC#;
2380 LPRINTTAB(40)AR;" / A/R" ;TAB(55)USINGL2$;GC#
2390 NEXT:CLOSE1:R1=R1+1:IFR1>32767THENR1=1
2400 LPRINTLF$:LPRINT"END OF REPORT":LPRINTLL$:LPRINTCHR$(12):GOTO1630
2410 LPRINTSTRING$(80,61)
2420 LPRINT"TYPE";TAB(9)"ACCT#";TAB(16)"NAME";TAB(32)"DATE";TAB(43)"INV#";TAB(60)
) "REMARKS"
2430 LPRINT"
";TAB(8)"AMOUNT (GLC)";TAB(26)"AMOUNT (GLC)";TAB(44)"AMOUNT (GLC)";TAB(63)"
TOTAL (GLC)":LPRINTSTRING$(80,61):RETURN
2440 LPRINTVA$;TAB(9)ABS(VB);TAB(16)L$;TAB(32)VC$;TAB(42)VD!;TAB(60)RM$
2450 LPRINT"* ";TAB(4)USINGL2$;VE#;:LPRINT"(";VJ;")";TAB(22)USINGL2$;VF#;:LPRINT
"(";VK;")";TAB(40)USINGL2$;VG#;:LPRINT"(";VL;")";TAB(58)USINGL2$;VH#;:LPRIN
T"(";A1;")":LPRINTLL$:RETURN
2460 CLS:IFU<>0THENPRINT@464,"UNPOSTED TRANSACTIONS IN SYSTEM":GOSUB3340:GOTO330
ELSEPRINT@468,"END OF PERIOD PROCESSING":GOSUB340:CLEAR50:RUN"PROCESS":END
2470 CLS:PRINT@476,"REPORTS":GOSUB340:CLEAR50:RUN"REPORTS":END
2480 JR=INT((J-1)/2)+1:JD=J-2*INT((J-1)/2)-1
2490 FIELD2,JD*127ASD$,11ASD0$,13ASD1$,17ASD2$,15ASD3$,5ASD4$,12ASD5$,4ASD6$,4AS
D7$,8ASD8$,8ASD9$,8ASDA$,8ASDB$,4ASDC$,2ASDD$,1ASDE$,1ASDF$,1ASDG$,1ASDH$,2
ASDI$,2ASDJ$:RETURN
2500 CB#=INT((CB#*100D0)+.5D0)/100D0:PB#=INT((PB#*100D0)+.5D0)/100D0:TS#=INT((TS
#*100D0)+.5D0)/100D0:PR#=INT((PR#*100D0)+.5D0)/100D0:RQ#=CHR$(RQ):ST#=CHR$(
ST):TC#=CHR$(TC):LSETD0$=F$:LSETD1$=L$:LSETD2$=SA$:LSETD3$=CS$:LSETD4$=Z$
2510 LSETD5$=PH$:LSETD6$=MKS$(DS!):LSETD7$=MKS$(DP!):LSETD8$=MKD$(PB#)
2520 LSETD9$=MKD$(TS#):LSETDA$=MKD$(PR#):LSETDB$=MKD$(CB#)
2530 LSETDC$=MKS$(CL!):LSETDD$=MKI$(ID):LSETDE$=RQ$
2540 LSETDF$=ST$:LSETDG$=FC$:LSETDH$=TC$:LSETDI$=MKI$(DI):LSETDJ$=MKI$(DJ)
2550 PUT2,JR:RETURN
2560 GET2,JR:RETURN
2570 F$=D0$:L$=D1$:SA$=D2$:CS$=D3$:Z$=D4$:PH$=D5$:DS!=CVS(D6$)
2580 DP!=CVS(D7$):PB#=CVD(DB$):TS#=CVD(D9$):PR#=CVD(DA$)
2590 CB#=CVD(DB$):CL!=CVS(DC$):ID=CVI(DD$):RQ$=DE$
2600 ST$=DF$:FC$=DG$:TC$=DH$:DI=CVI(DI$):DJ=CVI(DJ$)
2610 RQ=ASC(RQ$):ST=ASC(ST$):TC=ASC(TC$)
2620 DX!=DS!:GOSUB1040:DS$=IN$:DX!=DP!:GOSUB1040:DP$=IN$:RETURN
2630 PRINTT0$:PRINTT1$:PRINTT2$:PRINTT3$:PRINTT4$:PRINTT5$:PRINTT6$:PRINTT7$:PRI
NTT8$:PRINTT9$:RETURN
```

```
-----
2640 PRINTA$:PRINTTB$:PRINTTX$:PRINTTD$:PRINTTE$:PRINTTF$
2650 IF0$="H"THEN2660ELSEPRINTTAB(18)"(8) STATUS OF ACCOUNT:"GOSUB2690:RETURN
2660 PRINT
      (8) STATUS OF ACCOUNT : ";IFST=0THENPRINTTG$ELSEIFST=1THENPRINTTH$ELSEIFST
      =2THENPRINTTI$ELSEIFST=3THENPRINTTJ$ELSEIFST=4THENPRINTTK$ELSEIFST=5THENPRI
      NTTL$ELSEIFST=6THENPRINTTM$ELSEIFST=7THENPRINTTN$
2670 PRINT
      ";STRING$(64,95):RETURN
2680 PRINT@576,CHR$(30);"SELECT ONE OF THE FOLLOWING STATUS NUMBERS:"
2690 PRINTTAB(3)"0 - ";TG$;TAB(32)"4 - ";TK$
2700 PRINTTAB(3)"1 - ";TH$;TAB(32)"5 - ";TL$
2710 PRINTTAB(3)"2 - ";TI$;TAB(32)"6 - ";TM$
2720 PRINTTAB(3)"3 - ";TJ$;TAB(32)"7 - ";TN$
2730 RETURN
2740 IFF=0THENP(J,0)=J:P(J,1)=J:P(J,2)=J:RETURN
2750 FORN=1TOTN-1
2760 IFSGN(P(N,0))=-1THENP(N,0)=ABS(P(N,0)):J=P(N,0):TN=TN-1:F=F-1:GOTO2780
2770 NEXT
2780 FORN=1TOTN:IFABS(P(N,1))=JTHENP(N,1)=ABS(J):GOTO2800
2790 NEXT
2800 FORN=1TOTN:IFABS(P(N,2))=JTHENP(N,2)=ABS(J):RETURN
2810 NEXT:RETURN
2820 NL=0:NH=TN
2830 KS=INT((NH-NL)/2):IFKS=0THENRETURNELSEN=KS+NL
2840 IFN=NHTHENNH=INT((NH-NL)/2)+NL:GOTO2830ELSEIFP(N,AM)<0THENN=N+1:GOTO2840
2850 J=ABS(P(N,AM)):GOSUB2480:GOSUB2560:DD=CVI(DD$)
2860 IFAM=1THEN2890
2870 IFDD=VAL(L$)THENL$=DD$:GOTO2920
2880 IFDD>VAL(L$)THENNH=N:GOTO2830ELSE2910
2890 IFLEFT$(D1$,LEN(L$))=L$THENL$=D1$:GOTO2920
2900 IFLEFT$(D1$,LEN(L$))>L$THENNH=N:GOTO2830
2910 NL=N:GOTO2830
2920 IFSGN(P(N,AM))=-1THENKS=0:RETURNELSEGOSUB2570:RETURN
2930 PRINT@151,ID$:PRINT@216,F$:PRINT@280,L$:PRINT@344,SA$:
2940 PRINT@408,CS$:PRINT@472,Z$:PRINT@536,PH$:PRINT@599,CL!:
2950 PRINT@663,RQ%;" %":PRINT@728,FC$:
2960 RETURN
2970 PRINT@88,USINGL2$;PB$:PRINT@152,USINGL2$;TS$:
2980 PRINT@219,DS$:PRINT@280,USINGL2$;PR$:
2990 PRINT@347,DP$:PRINT@408,USINGL2$;CB$:
3000 IF0$="H"THENRETURNELSEPRINT@VAL(MID$(SS$,ST*3+1,3))-192,"->":RETURN
3010 KR=INT((K-1)/8)+1:KD=K-8*INT((K-1)/8)-1
3020 FIELD3,KD*31ASVV$,1ASV0$,2ASV1$,4ASV2$,4ASV3$,4ASV4$,4ASV5$,4ASV6$,1ASV7$,2
      ASV8$,2ASV9$,1ASVX$,1ASVY$,1ASVZ$:RETURN
3030 LSETV0$=VA$:LSETV1$=MKI$(VB):LSETV2$=MKS$(VC!):LSETV3$=MKS$(VD!):LSETV4$=VE
      $:LSETV5$=VF$:LSETV6$=VG$:LSETV7$=VI$:LSETVX$=VJ$:LSETVY$=VK$:LSETVZ$=VL$
3040 PUT3,KR:RETURN
3050 GET3,KR:RETURN
3060 VA$=V0$:VB=CVI(V1$):VC!=CVS(V2$):VD!=CVS(V3$):VE$=V4$:VF$=V5$:VG$=V6$:VI$=V
      7$:VJ=ASC(VX$):VK=ASC(VY$):VL=ASC(VZ$):VJ$=VX$:VK$=VY$:VL$=VZ$
```

```

3070 DX!=VC!:GOSUB1040:VC$=IN$:V$=VE$:GOSUB3860:VE#=N#:V$=VF$:GOSUB3860:VF#=N#:V
    $=VG$:GOSUB3860:VG#=N#:VH#=VE#+VF#+VG#:RETURN
3080 GR=INT((G-1)/25)+1:GD=G-25*INT((G-1)/25)-1
3090 FIELD1,GD*10ASG0$,2ASG1$,8ASG3$:RETURN
3100 LSETG1$=MKI$(GA):LSETG3$=MKD$(GC#)
3110 PUT1,GR:RETURN
3120 GET1,GR:RETURN
3130 GA=CVI(G1$):GC#=CVD(G3$):RETURN
3140 CLS:PRINTTAB(20)"CUSTOMER ACCOUNTS
    ":GOSUB2630:GOSUB2930:PT=2:RETURN
3150 CLS:PRINT@64,CHR$(31):GOSUB2640:GOSUB2970:PT=1
3160 PRINTCHR$(28);"ACCOUNT:";ID;"- ";L$;,"PAGE 2":RETURN
3170 PRINT@768,STRING$(64,95);"(S)EARCH          (P)REVIOUS          (N)EXT          (T)U
    RN PAGE
    (U)PDATE          (H)ARD COPY          (D)ELETE          (Q) EXIT TO MENU
    ENTER SELECTION --> ";
3180 RETURN
3190 PRINTCHR$(29);CHR$(30):RETURN
3200 IFPEEK(14312)<128THENPOKE16425,1:RETURNELSECLS:PRINT@468,"PRINTER NOT READY
    ":GOSUB3340:CF=2:RETURN
3210 IFPEEK(16425)<54THENRETURNELSECF=2:LPRINTCHR$(12):RETURN
3220 CLS:PRINT@468,"ENTER PASSWORD --> ";
3230 P$="":FORX=1TO8
3240 PE$=INKEY$:IFPE$=""THEN3240ELSEIFPE$<" "ORPE$>"Z"THEN3260
3250 PRINT"#":P$=P$+PE$:GOTO3270
3260 IFASC(PE$)=13THEN3280ELSE3240
3270 NEXT
3280 PI$="CUSINDEX."+P$+":1":PD$="CUSDATA."+P$+":1"
3290 PS$="CUSSETUP."+P$+":1":PT$="TRANSACTION."+P$+":1"
3300 PG$="GLFILE."+P$+":1"
3310 RETURN
3320 PRINT"
    TYPE @ TO EXIT --> ";
3330 FL=-1:GOSUB390:IFCF=1THENRETURNELSEPRINTCHR$(8):GOTO3330
3340 FORTT=1TO2500:NEXT:RETURN
3350 OPEN"I",1,PS$
3360 LINEINPUT#1,N$:LINEINPUT#1,A$:LINEINPUT#1,CZ$
3370 LINEINPUT#1,PN$:INPUT#1,FC!,GT,PD,PC,PT,PP:LINEINPUT#1,GL$
3380 CLOSE1:DIMG$(GT):IFPD=2THENPT$=LEFT$(PT$,LEN(PT$)-1)+"2"
3390 OPEN"I",1,PI$
3400 INPUT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7,CV#
3410 FORN=1TOTN-1:INPUT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1
3420 OPEN"R",2,PD$:RETURN
3430 OPEN"O",1,PI$
3440 PRINT#1,TI;TN;F;U;UT;UN;EP;R0;R1;R2;R3;R4;R5;R6;R7;STR$(CV#)+"D0"
3450 FORN=1TOTN-1:PRINT#1,P(N,0);P(N,1);P(N,2):NEXT:CLOSE1:RETURN
3460 CLS:PRINTTAB(20)"SYSTEM STATUS":PRINT@117,LEFT$(TIME$,8)
3470 PRINTTAB(20)N$:PRINTTAB(20)A$:PRINTTAB(20)CZ$;"
    "
3480 PRINT"NUMBER OF CUSTOMERS ";STRING$(6,46)USINGL0$;TI

```

```
-----
3490 PRINT"TRANSACTIONS
      POSTED ";STRING$(13,46);USINGL0$;UT-U
3500 PRINT"      UNPOSTED ";STRING$(11,46);USINGL0$;U
3510 PRINT"      TOTAL ";STRING$(14,46);USINGL0$;UT
3520 PRINT"
      AMOUNT RECEIVABLE ";STRING$(5,46);USINGL2$;CV#
3530 PRINT"FINANCE CHARGE ";STRING$(14,46);USINGL3$;FC!*100;:PRINT" %":GOSUB3320
      :GOTO3330
3540 IFERR/2+1=54THENCLS:PRINT@465,"NO FILES ON DATA DISK

      ":CLEAR50:END
3550 IFERR/2+1=57THEN3580
3560 IFERR/2+1=70THENCLS:PRINT@465,"*** INVALID PASSWORD ***

      PROGRAM TERMINATED

      ":CLEAR50:END
3570 PRINT:PRINTCHR$(30);"UNEXPECTED ERROR CODE ";ERR/2+1;"IN LINE";ERL:GOSUB332
0:GOTO370
3580 TE=TE+1:IFTE>5THENCLS:PRINT@404,"** DISK I/O ERROR **":GOSUB3320:GOTO370ELS
ERESUME
3590 DATA"####", "####.##", "#####.##+", "###.##"
3600 DATA"ACCOUNT NUMBER ----->","FIRST NAME, INITIAL->"
3610 DATA"LAST NAME ----->","(1) STREET ADDRESS ->"
3620 DATA"(2) CITY, STATE ----->","(3) ZIP CODE ----->"
3630 DATA"(4) PHONE NUMBER ---->","(5) CREDIT LIMIT ---->"
3640 DATA"(6) MIN. PAYMENT % ->","(7) FIN. CHG. (Y/N)->"
3650 DATA"PREVIOUS BALANCE ---->","SALES THIS MONTH ---->"
3660 DATA"DATE OF LAST SALE -->","PAYMENTS RECEIVED -->"
3670 DATA"DATE LAST PAYMENT -->","CURRENT BALANCE ---->"
3680 DATA"NOTHING DUE / CREDIT"
3690 DATA"CURRENT BALANCE DUE"
3700 DATA"30-60 DAYS OVERDUE"
3710 DATA"60-90 DAYS OVERDUE"
3720 DATA"PAST 90 DAYS OVERDUE"
3730 DATA"CREDIT SUSPENDED"
3740 DATA"LEGAL ACTION PENDING"
3750 DATA"INACTIVE ACCOUNT"
3760 DATA"704768832896733797861925"
3770 DATA33,64,60,6,64,78,62,127,145,48,2,14,32,121,205,59
3780 DATA0,35,16,241,62,13,205,59,0,124,254,63,32,229,125
3790 DATA254,0,32,224,201
3800 N#=VAL(IN$):W#=ABS(N#)*100:V$=""
3810 X=W#/D1#:W#=W#-X*D1#:V$=V$+CHR$(X-(N#<0)*128)
3820 X=W#/D2#:W#=W#-X*D2#:V$=V$+CHR$(X)
3830 X=W#/D3#:W#=W#-X*D3#:V$=V$+CHR$(X)
3840 V$=V$+CHR$(W#)
3850 RETURN
3860 W#=(ASC(MID$(V$,1,1))AND127)*D1#+ASC(MID$(V$,2,1))*D2#+ASC(MID$(V$,3,1))*D3
#+ASC(MID$(V$,4,1))
3870 IFASC(LEFT$(V$,1))>127THENW#=-W#
```

```
3880 N#=W#/100
3890 RETURN
3900 IFPEEK(14312)>127THENRETURN
3910 SV!=(PEEK(VARPTR(SV$)+1))+256*(PEEK(VARPTR(SV$)+2))
3920 IFSV!>32767THENSV!=SV!-65536
3930 DEFUSR0=SV!:SV=USR(0):RETURN
```

```
450 GOSUB1080
460 GOSUB320:K=UN
470 IFK>PTTHENPRINT@128,"FILE IS FULL (";PT;"RECORDS)":GOSUB1170:GOTO120
480 PRINT
490 PRINT@256,"(N)ORMAL OR (C)ORRECTION ENTRY ? ";
500 FL=1:GOSUB130:IFCF<>0THENGOTO120
510 IFIN$<>"N"ANDIN$<>"C"THENPRINTCHR$(8):GOTO500ELSEVI$=IN$
520 PRINT:GOSUB330
530 PRINT@320,"(S)ALE, (P)AYMENT OR (B)AD DEBT ? ";
540 FL=1:GOSUB130:IFCF<>0THEN1030
550 IFIN$<>"S"ANDIN$<>"P"ANDIN$<>"B"THENPRINTCHR$(8):GOTO540ELSEVA$=IN$
560 PRINT
570 PRINT@448,"CUSTOMER ACCOUNT NUMBER: ";
580 FL=-4:GOSUB130:IFCF<>0THEN1030ELSEIFIN$<>" "THENVB=VAL(IN$)ELSE570
590 PRINT@480,"* CHECKING FOR VALID ACCOUNT # *":GOSUB1420:PRINT@480,CHR$(30);:
    IFKS=0THENIN$="INVALID ACCOUNT NUMBER":GOSUB420:GOTO570
595 IFVA$="B"ANDCVD(DB$)<=0THENIN$="CUST. HAS CREDIT BALANCE":GOSUB420:GOTO1030
600 PRINT@512,"DATE (PRESS <ENTER> TO USE TODAY'S DATE): ";
610 FL=8:GOSUB130:IFCF<>0THEN1030ELSEIFIN$=" "THENGOSUB1060
620 GOSUB370:IFCF=2THEN600ELSEVC!=DX!
630 PRINT@576,"CUSTOMER INVOICE NUMBER: ";
640 FL=-6:GOSUB130:IFCF<>0THEN1030ELSEIFIN$<>" "THENVD!=VAL(IN$)ELSE630
650 IFVA$="P"ORVA$="B"THENB60
660 PRINT@640,"GLC: ";
670 FL=-2:GOSUB130:IFCF<>0THEN1030ELSEIFVAL(IN$)>4ANDVAL(IN$)<=GTTHENVJ$=CHR$(VAL(IN$))ELSE660
680 PRINT@650,":GOSUB1050
690 PRINT@670,CHR$(30);"AMOUNT: ";
700 FL=-10:GOSUB130:GOSUB340:IFCF=0THENVE#=VAL(IN$):PRINTCHR$(30);ELSEONCFGOTO1030,690
710 PRINT@704,"GLC: ";
720 FL=-2:GOSUB130:IFCF<>0THEN1030ELSEIFIN$=" "THENVK$=CHR$(0):VF#=0:GOTO770
730 IFVAL(IN$)>4ANDVAL(IN$)<=GTTHENVK$=CHR$(VAL(IN$))ELSE710
740 PRINT@714,":GOSUB1050
750 PRINT@734,CHR$(30);"AMOUNT: ";
760 FL=-10:GOSUB130:GOSUB340:IFCF=0THENVF#=VAL(IN$):PRINTCHR$(30);ELSEONCFGOTO1030,750
770 PRINT@768,"GLC: ";
780 FL=-2:GOSUB130:IFCF<>0THEN1030ELSEIFIN$=" "THENVL$=CHR$(0):VG#=0:GOTO830
790 IFVAL(IN$)>4ANDVAL(IN$)<=GTTHENVL$=CHR$(VAL(IN$))ELSE770
800 PRINT@778,":GOSUB1050
810 PRINT@798,CHR$(30);"AMOUNT: ";
820 FL=-10:GOSUB130:GOSUB340:IFCF=0THENVG#=VAL(IN$):PRINTCHR$(30);ELSEONCFGOTO1030,810
830 IN$="1":VH#=VE#VF#VG#:PRINT@832,"GLC: 1":PRINT@842,;
840 GOSUB1050:PRINT@856,"TOTAL AMOUNT: ";
850 PRINTUSINGL2$;VH#:GOTO920
860 PRINT@640,"GLC: ";:IFVA$="B"THENPRINT"4":VJ$=CHR$(4):IN$="4":GOTO880
870 FL=-2:GOSUB130:IFCF<>0THEN1030ELSEIFVAL(IN$)>4ANDVAL(IN$)<=GTTHENVJ$=CHR$(VAL(IN$))ELSE860
```

```

880 PRINT@650,;:GOSUB1050
890 IFVA$="P"THENPRINT@670,CHR$(30);"AMOUNT OF PAYMENT: ";ELSEPRINT@670,CHR$(30)
    ;"AMOUNT OF BAD DEBT: ";CVD(DB$):VE#=-CVD(DB$):LSETDF$=CHR$(5):GOTO910
900 FL=-10:GOSUB130:GOSUB340:IFCF=0THENVE#=-VAL(IN$):PRINTCHR$(30);ELSEONCFGOTO1
    030,890
910 VF#=0:VG#=0:VK$=CHR$(0):VL$=CHR$(0)
920 PRINT@960,CHR$(30);TAB(20)"RECORDING THIS TRANSACTION":;IFVI$="N"THEN1020
930 FORN=1TOUN-1
940 K=N:GOSUB1090:GOSUB1130
950 IFABS(CVI(V1$))<>VBORCVS(V3$)<>VD!ORV7$="A"THEN990
960 IFSGN(CVI(V1$))=1THENLSETV7$="A":GOTO980
970 LSETV7$="B"
980 PUT3,KR
990 NEXT:K=UN:GOTO1020
1000 IN$=STR$(VE#):GOSUB1330:VE$=V$:IN$=STR$(VF#):GOSUB1330:VF$=V$
1010 IN$=STR$(VG#):GOSUB1330:VG$=V$:RETURN
1020 GOSUB1000:GOSUB1070:GOSUB1140
1030 GOSUB440:GOSUB1040:GOTO460
1040 PRINT@960,"( LAST ENTRY: ACCT.# :":Z1:"/ INV.# :":Z2!:"/ # OF TRANS.":UT;)"
    ":RETURN
1050 PRINTMID$(GL$(VAL(IN$)-1)*5+1,5);:RETURN
1060 IN$=LEFT$(TIME$,8):PRINTIN$;:RETURN
1070 GOSUB1500:GOSUB1090:GOSUB1130:GOSUB1110:U=U+1:UT=UT+1:UN=UN+1:Z1=VB:Z2!=VD!
    :RETURN
1080 OPEN"R",3,PT$:OPEN"R",2,PD$:RETURN
1090 KR=INT((K-1)/8)+1:KD=K-8*INT((K-1)/8)-1
1100 FIELD3,KD*31ASVV$,1ASV0$,2ASV1$,4ASV2$,4ASV3$,4ASV4$,4ASV5$,4ASV6$,1ASV7$,2
    ASV8$,2ASV9$,1ASVX$,1ASVY$,1ASVZ$:RETURN
1110 LSETV0$=VA$:LSETV1$=MKI$(VB):LSETV2$=MKS$(VC!):LSETV3$=MKS$(VD!):LSETV4$=VE
    $:LSETV5$=VF$:LSETV6$=VG$:LSETV7$=VI$:LSETV8$=MKI$(KI):LSETV9$=MKI$(KJ):LSE
    TVX$=VJ$:LSETVY$=VK$:LSETVZ$=VL$
1120 PUT3,KR:RETURN
1130 GET3,KR:RETURN
1140 PRINT@256,CHR$(31);:RETURN
1150 PRINT:PRINT"TYPE @ TO EXIT --> ";
1160 FL=-1:GOSUB130:IFCF=1THENRETURNELSEPRINTCHR$(8);:GOTO1160
1170 FORTT=1TO2500:NEXT:RETURN
1180 OPEN"I",1,PS$
1190 LINEINPUT#1,N$:LINEINPUT#1,A$:LINEINPUT#1,CZ$
1200 LINEINPUT#1,PN$:INPUT#1,FC!,GT,PD,PC,PT,PP:LINEINPUT#1,GL$
1210 CLOSE1:DIMG#(GT)
1220 OPEN"I",1,PI$
1230 INPUT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7,CV#
1240 FORN=1TOTN-1:INPUT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1:RETURN
1250 OPEN"O",1,PI$
1260 PRINT#1,TI;TN;F;U;UT;UN;EP;R0;R1;R2;R3;R4;R5;R6;R7;STR$(CV#)+"D0"
1270 FORN=1TOTN-1:PRINT#1,P(N,0);P(N,1);P(N,2):NEXT:CLOSE1:RETURN
1280 IFERR/2+1=54THENCLS:PRINT@465,"NO FILES ON DATA DISK":PRINT:PRINT:CLEAR50:E
    ND
1290 IFERR/2+1=57THEN1320

```

```
=====
1300 IFERR/2+1=70THENCLS:PRINT@465,"*** INVALID PASSWORD ***":PRINT:PRINTTAB(20)
    "PROGRAM TERMINATED":PRINT:CLEAR50:END
1310 PRINT:PRINTCHR$(30);"UNEXPECTED ERROR CODE ";ERR/2+1;"IN LINE":ERL:GOSUB115
    0:GOTO120
1320 TE=TE+1:IFTE>5THENCLS:PRINT@404,"** DISK I/O ERROR **":GOSUB1150:GOTO120ELSE
    ERESUME
1330 N#=VAL(IN$):W#=ABS(N#)*100:V$=""
1340 X=W#/D1#:W#=W#-X*D1#:V$=V$+CHR$(X-(N#<0)*128)
1350 X=W#/D2#:W#=W#-X*D2#:V$=V$+CHR$(X)
1360 X=W#/D3#:W#=W#-X*D3#:V$=V$+CHR$(X)
1370 V$=V$+CHR$(W#)
1380 RETURN
1390 W#=(ASC(MID$(V$,1,1))AND127)*D1#+ASC(MID$(V$,2,1))*D2#+ASC(MID$(V$,3,1))*D3
    #+ASC(MID$(V$,4,1))
1400 IFASC(LEFT$(V$,1))>127THENW#=-W#
1410 N#=W#/100:RETURN
1420 NL=0:NH=TN
1430 KS=INT((NH-NL)/2):IFKS=0THENRETURNELSEN=KS+NL
1440 IFN=NHTHENNH=INT((NH-NL)/2)+NL:GOTO1430ELSEIFP(N,0)<0THENN=N+1:GOTO1440
1450 J=ABS(P(N,0)):GOSUB1480:DD=CVI(DD$)
1460 IFDD=VBTHENRETURN
1470 IFDD>VBTHENNH=N:GOTO1430ELSENL=N:GOTO1430
1480 JR=INT((J-1)/2)+1:JD=J-2*INT((J-1)/2)-1:FIELD2,JD*127ASD$,11ASD0$,13ASD1$,1
    7ASD2$,15ASD3$,5ASD4$,12ASD5$,4ASD6$,4ASD7$,8ASD8$,8ASD9$,8ASDA$,8ASDB$,4AS
    DC$,2ASDD$,1ASDE$,1ASDF$,1ASDG$,1ASDH$,2ASDI$,2ASDJ$
1490 GET2,JR:RETURN
1500 DI=CVI(DI$):DJ=CVI(DJ$):KI=DI:KJ=DJ:KK=K:IFDJ=0THENDJ=K:DI=K:KI=0:KJ=0:LSET
    DI$=MKI$(DI):LSETDJ$=MKI$(DJ):PUT2,JR:RETURN
1510 K=DJ:GOSUB1090:GOSUB1130:LSETV9$=MKI$(KK):GOSUB1120:KI=DJ:KJ=0:DJ=KK:K=KK:L
    SETDJ$=MKI$(DJ):PUT2,JR:RETURN
```

```

10 REM ** ARS REPORT PRINTING PROGRAM **
20 REM COPYRIGHT (C) 1979, TANDY CORP.
30 REM VERSION 1.0 - 07/16/79
40 CLEAR600:DEFINT A-Z:DIMP(500,2),AN(8),AN#(8),W,WL,WS,WD,W$,CF,FL,IN$:W$=""
50 ONERRORGOTO1820:OPEN"I",1,"TRANSFER"
60 LINEINPUT#1,PI$:LINEINPUT#1,PD$
70 LINEINPUT#1,PS$:LINEINPUT#1,PT$:LINEINPUT#1,PG$
80 CLOSE1
90 POKE16424,67:POKE16425,1
100 READL0$,L1$,L2$,L3$,L4$,L5$,TG$,TH$,TI$,TJ$,TK$,TL$,TM$,TN$,SS$:VS$="":FORX=
    1TO40:READVS:VS$=VS$+CHR$(VS):NEXT:RESTORE
110 XX=1:LF$=" ":LL$=STRING$(80,"-")
120 GOSUB1710:J=1:GOSUB1480:GOSUB380
130 CLS:PRINT@468,"RETURNING TO MAIN MENU":GOSUB1790:CLOSE:CLEAR50:RUN"ARS":END
140 W=FRE(W$):IN$="":CF=0:W$=INKEY$:WD=0:WS=WD:WL=WD:IFFL=WDTHENFL=1
150 PRINTSTRING$(ABS(FL),CHR$(136)):STRING$(ABS(FL),CHR$(24)):
160 PRINTCHR$(14):FORW=1TO25:W$=INKEY$:IFW$<>" "THEN170ELSENEXT:PRINTCHR$(15):F
    ORW=1TO25:W$=INKEY$:IFW$<>" "THEN170ELSENEXT:GOTO160
170 IFW$<>CHR$(13)THEN190ELSEPRINTSTRING$(ABS(FL)-WL," "):
180 PRINTCHR$(15):W=25:NEXT:RETURN
190 IFW$<>"@"THEN210
200 CF=1:PRINTCHR$(15):RETURN
210 PRINTCHR$(14):IFW$=CHR$(24)THENPRINTSTRING$(WL,CHR$(24)):GOTO140
220 IFW$<>CHR$(8)THEN260ELSEIFWL=0THEN160ELSEPRINTCHR$(24):IFFL>0THEN240ELSEIFP
    EEK(16418)=44THEN250
230 IFPEEK(16418)=46THENWD=0:GOTO240ELSEIFPEEK(16418)=43ORPEEK(16418)=45THENWS=0
240 IN$=LEFT$(IN$,LEN(IN$)-1)
250 WL=WL-1:POKE16418,136:GOTO160
260 IFABS(FL)=WLTHEN160ELSEIFFL>0THENIFW$>=" "ANDW$<="z"THEN310
270 IFW$=","ANDWD=0THENWD=1:GOTO310
280 IFW$=","THENPRINTW$:WL=WL+1:GOTO320
290 IF(W$="-"ORW$="+")ANDWS=0ANDWL=0THENWS=1:GOTO310
300 IFW$<"0"ORW$>"9"THEN160
310 PRINTW$:IN$=IN$+W$:WL=WL+1
320 IFABS(FL)=1THEN180ELSE160
330 PRINT@128,CHR$(30):"(TYPE @ TO EXIT TO MAIN MENU)":RETURN
340 DX$=STR$(DX!):DX$=RIGHT$(DX$,LEN(DX$)-1)
350 IFLEN(DX$)<6THENFORW=1TO6-LEN(DX$):DX$="0"+DX$:NEXT
360 IN$=MID$(DX$,3,2)+"/"+MID$(DX$,5,2)+"/"+MID$(DX$,1,2)
370 RETURN
380 CLS:PRINTTAB(29)"REPORTS":PRINT
390 PRINT"
    PRINT OPTIONS:

    NOT(E)NTIRE ACCOUNT RECORDS
    NOT(S)TATUS AGING CATEGORIES
    NOT(A)GING STATUS ANALYSIS
    NOT(C)URRENT BALANCE RECORDS"
400 PRINT"/(OVER CREDIT LIMIT)

```

```
-----
NOT( @ ) EXIT TO MAIN MENU"
410 PRINT"

    SELECT OPTION --> ";
420 FL=1:GOSUB140:IFCF<>0THENRETURN
430 O=INSTR("ESCA",IN$):IF0=0THEN380ELSEONOGOTO440,670,990,780
440 GOSUB1660:IFCF=2THEN380
450 GOSUB1060:IFCF=1THEN380
460 GOSUB1240:IFCF=1THEN380
470 LPRINTLEFT$(TIME$,8);" - COMPLETE ACCOUNT LISTING - RUN#";R2
480 LPRINTLF$:LPRINTSQ$
490 GOSUB500:R2=R2+1:IFR2>32767THENR2=1:GOTO380ELSE380
500 ONRSGOSUB1270,1310
510 RT#=0
520 FORN=1TOTN-1
530 GOSUB1670:IFCF=2THENONRSGOSUB1270,1310
540 J=P(N,AM):IFSGN(J)=-1THEN610ELSEGOSUB1480:GOSUB1500
550 IF0=1THEN590
560 IF0=2THEN580
570 IFCVD(DB$)<=CVS(DC$)THEN610ELSE590
580 ST=ASC(DF$):IFST<S1ORST>S2THEN610
590 GOSUB1510:ONRSGOSUB1300,1350:RT#=RT#+CB#
600 EN$=INKEY$:IFEN$="@"THENLPRINT"RUN TERMINATED":N=TN-1:NEXT:RETURN
610 NEXT:LPRINTLF$
620 LPRINT"RECEIVABLES FOR THESE RECORDS: ";USINGL2$;RT#
630 IFCV#=0THEN660
640 RP!=RT#/CV#*100
650 LPRINT"THIS IS ";USINGL5$;RP!";LPRINT" % OF ";USINGL2$;CV#";LPRINT" (TOTAL
    RECEIVABLES). "
660 LPRINTLL$:LPRINT"END OF REPORT":LPRINTCHR$(12):RETURN
670 GOSUB1660:IFCF=2THEN380
680 GOSUB1060:IFCF=1THEN380
690 GOSUB1240:IFCF=1THEN380
700 GOSUB1110:IFCF=1THEN380
710 LPRINTLEFT$(TIME$,8);" - LISTING BY AGING STATUS - RUN#";R3
720 LPRINTLF$
730 ONS1+1GOSUB1400,1410,1420,1430,1440,1450,1460,1470
740 LPRINTSV$:IFS2=S1THEN760ELSELPRINT"      TO"
750 ONS2+1GOSUB1400,1410,1420,1430,1440,1450,1460,1470:LPRINTSV$
760 LPRINTLF$:LPRINTSQ$
770 GOSUB500:R3=R3+1:IFR3>32767THENR3=1:GOTO380ELSE380
780 FORX=0TO8:AN(X)=0:AN#(X)=0:NEXT:TA=0:TA#=0:TP!=0
790 CLS:PRINT@468,"SCANNING ACCOUNT RECORDS":FORN=1TOTN-1
800 J=P(N,0):IFSGN(J)=-1THEN830
810 GOSUB1480:GOSUB1500:ST=ASC(DF$)
820 AN#(ST)=AN#(ST)+CVD(DB$):AN(ST)=AN(ST)+1
830 NEXT
840 CLS:PRINT"ACCOUNTS RECEIVABLE ANALYSIS BY AGING STATUS"
850 PRINT
860 PRINT"AGING CATEGORY";TAB(31)"ACCTS";TAB(41)"AMOUNT";TAB(52)"% OF A/R"
```

```
870 PRINT"-----";TAB(31)"-----";TAB(41)"-----";TAB(52)"-----"
880 FORX=0TO7:TA=TA+AN(X):TA#=TA#+AN#(X):NEXT:FORX=0TO7
890 PC!=AN#(X)/TA#*100:TP!=TP!+PC!
900 ONX+1GOSUB1400,1410,1420,1430,1440,1450,1460,1470
910 PRINTSV$;TAB(30)USINGL1$;AN(X);:PRINTTAB(37)USINGL2$;AN#(X);:PRINTTAB(50)USI
    NGL5$;PC!;:PRINT" %"
920 NEXT:PRINTTAB(31)"-----";TAB(41)"-----";TAB(52)"-----"
930 PRINTTAB(20)"TOTALS";TAB(30)USINGL1$;TA;:PRINTTAB(37)USINGL2$;TA#;:PRINTTAB(
    50)USINGL5$;TP!;:PRINT" %"
940 PRINT@970,"DO YOU WANT A PRINTOUT ? (Y/N) ";
950 FL=1:GOSUB140:IFCF<>0THEN380ELSEIFIN$<>"Y"ANDIN$<>"N"THENPRINTCHR$(8);:GOTO9
    50
960 IFIN$="N"THEN380ELSEIFPEEK(14312)>127THENGOSUB1650:PRINT@970,"P R I N T E R
    N O T   R E A D Y   ! !";:GOSUB1700:GOSUB1650:GOTO940
970 GOSUB1650:PRINT@970,LEFT$(TIME$,8);" - REPORT #";R5;
980 GOSUB1890:R5=R5+1:IFR5>32767THENR5=1:GOTO380ELSE380
990 GOSUB1660:IFCF=2THEN380
1000 GOSUB1060:IFCF=1THEN380
1010 GOSUB1240:IFCF=1THEN380
1020 LPRINTLEFT$(TIME$,8);" - RECORDS WITH BALANCE OVER THE CREDIT LIMIT - RUN#"
    ;R4
1030 LPRINTLF$
1040 LPRINTSQ$
1050 GOSUB500:R4=R4+1:IFR4>32767THENR4=1:GOTO380ELSE380
1060 CLS:PRINT"PRINT BY :
```

(A)CCOUNT NUMBER

(N)AME

(Z)IP CODE

ENTER SELECTION --> " ;

```
1070 FL=1:GOSUB140:IFCF<>0THENRETURN
1080 IFIN$<>"A"ANDIN$<>"N"ANDIN$<>"Z"THENPRINTCHR$(8);:GOTO1070
1090 IFIN$="A"THENAM=0:SQ$="(ACCT# SEQUENCE)"ELSEIFIN$="N"THENAM=1:SQ$="(NAME SE
    QUENCE)"ELSEAM=2:SQ$="(ZIP CODE SEQUENCE)"
1100 PRINT:RETURN
1110 CLS:PRINT@192,"SELECT 2 NUMBERS FROM THE TABLE BELOW. THE ARS WILL PRINT AL
    L
    RECORDS WITH STATUS BETWEEN AND INCLUDING THE 2 NUMBERS. TO
    SELECT ONLY 1 STATUS CODE, TYPE THE SAME NUMBER TWICE.":GOSUB1570
1120 PRINT@448,"FIRST NUMBER : ";
1130 FL=-1:GOSUB140:IFCF<>0THENRETURN
1140 S1=VAL(IN$):IFS1<0ORS1>7THENPRINTCHR$(8);:GOTO1130
1150 ST=S1:GOSUB1630
1160 PRINT@512,"SECOND NUMBER: ";
1170 FL=-1:GOSUB140:IFCF<>0THENRETURN
1180 S2=VAL(IN$):IFS2<0ORS2>7THENPRINTCHR$(8);:GOTO1170
1190 ST=S2:GOSUB1630
```

```

1200 PRINT@576,"OK TO PRINT (Y/N) ? ";
1210 FL=1:GOSUB140:IFCF<>0THENRETURNELSEIFIN<>"Y"ANDIN<>"N"THENPRINTCHR$(8):;G
    OT01210
1220 IFIN$="N"THEN1110
1230 IFS1>S2THENS3=S1:S1=S2:S2=S3:RETURNELSERETURN
1240 PRINT"(A)BBREVIATED OR (F)ULL RECORDS ? (A/F) ";
1250 FL=1:GOSUB140:IFCF<>0THENRETURN
1260 IFIN<>"A"ANDIN<>"F"THENPRINTCHR$(8):;GOTO1250ELSEIFIN$="A"THENRS=1:RETURN
    ELSERS=2:RETURN
1270 LPRINTSTRING$(80,61)
1280 LPRINT"ACCT#";TAB(6)"LAST NAME";TAB(21)"CRT LMT";TAB(30)"PRV BAL";TAB(42)"T
    TL SLS";TAB(54)"PMT REC";TAB(63)"    CUR BAL STATUS"
1290 LPRINTSTRING$(80,61):CF=0:RETURN
1300 LPRINTUSINGL1$;ID::LPRINTTAB(6)L$;TAB(19)USINGL4$;CL!:;LPRINTTAB(26)USINGL2
    $;PB#:;LPRINTTAB(38)USINGL2$;TS#:;LPRINTTAB(50)USINGL2$;PR#:;LPRINTTAB(62)U
    SINGL2$;CB#:;LPRINT"    ";USINGL0$;ST:RETURN
1310 LPRINTSTRING$(80,61)
1320 LPRINT"ACCT#";TAB(10)"NAME OF ACCOUNT";TAB(33)"ADDRESS";TAB(51)"CITY, STATE
    ZIP":LPRINTSTRING$(80,61)
1330 LPRINT"PHONE";TAB(15)"CRT LMT";TAB(29)"%RQ";TAB(39)"FC?";TAB(49)"DT LAST SA
    LE";TAB(63)"    DT LAST PYMT"
1340 LPRINT"
    PRV BAL";TAB(18)"TTL SLE";TAB(32)"PMT RCV";TAB(45)"CUR BAL";TAB(57)"
    STATUS":LPRINTSTRING$(80,61):LPRINT"    ":CF=0:RETURN
1350 IFRIGHT$(F$,1)<>CHR$(0)THEN1370ELSEIFASC(F$)=0THENF$=CHR$(32):GOTO1370
1360 F$=LEFT$(F$,INSTR(F$,CHR$(0))-1)
1370 LPRINTUSINGL1$;ID::LPRINTTAB(8)F$;TAB(19)L$;TAB(33)SA$;TAB(51)CS$;TAB(63)"
    ";Z$
1380 LPRINTSTRING$(80,61):LPRINTPH$;TAB(15)USINGL4$;CL!:;LPRINTTAB(28)USINGL1$;
    RQ::LPRINTTAB(40)FC$;TAB(51)DS$;TAB(63)"    ";DP$
1390 LPRINTUSINGL2$;PB#:;LPRINTTAB(14)USINGL2$;TS#:;LPRINTTAB(28)USINGL2$;PR#:;L
    PRINTTAB(41)USINGL2$;CB#:;ONST+1GOSUB1400,1410,1420,1430,1440,1450,1460,147
    0:LPRINTTAB(56)SV$:LPRINTLL$:LPRINT"    ":RETURN
1400 SV$="0 - "+TG$:RETURN
1410 SV$="1 - "+TH$:RETURN
1420 SV$="2 - "+TI$:RETURN
1430 SV$="3 - "+TJ$:RETURN
1440 SV$="4 - "+TK$:RETURN
1450 SV$="5 - "+TL$:RETURN
1460 SV$="6 - "+TM$:RETURN
1470 SV$="7 - "+TN$:RETURN
1480 JR=INT((J-1)/2)+1:JD=J-2*INT((J-1)/2)-1
1490 FIELD2,JD*127ASD$,11ASD0$,13ASD1$,17ASD2$,15ASD3$,5ASD4$,12ASD5$,4ASD6$,4AS
    D7$,8ASD8$,8ASD9$,8ASDA$,8ASDB$,4ASDC$,2ASDD$,1ASDE$,1ASDF$,1ASDG$,1ASDH$,2
    ASDI$,2ASDJ$:RETURN
1500 GET2,JR:RETURN
1510 F$=D0$:L$=D1$:SA$=D2$:CS$=D3$:Z$=D4$:PH$=D5$:DS!=CVS(D6$)
1520 DP!=CVS(D7$):PB#=CVD(D8$):TS#=CVD(D9$):PR#=CVD(DA$)
1530 CB#=CVD(DB$):CL!=CVS(DC$):ID=CVI(DD$):RQ$=DE$
1540 ST$=DE$:FC$=DG$:TC$=DH$:DI=CVI(DI$):DJ=CVI(DJ$)

```

```

1550 RQ=ASC(RQ$):ST=ASC(ST$):TC=ASC(TC$)
1560 DX!=DS!:GOSUB340:DS%=IN$:DX!=DP!:GOSUB340:DP%=IN$:RETURN
1570 PRINT@704,;
1580 PRINTTAB(3)"0 - ";TG$;TAB(32)"4 - ";TK$
1590 PRINTTAB(3)"1 - ";TH$;TAB(32)"5 - ";TL$
1600 PRINTTAB(3)"2 - ";TI$;TAB(32)"6 - ";TM$
1610 PRINTTAB(3)"3 - ";TJ$;TAB(32)"7 - ";TN$
1620 RETURN
1630 PRINT@VAL(MID$(SS$,ST*3+1,3)),"->";:RETURN
1640 PRINT@256,CHR$(31);:PRINT@256,;:RETURN
1650 PRINTCHR$(29);CHR$(30);:RETURN
1660 IFPEEK(14312)<128THENPOKE16425,1:RETURNELSECLS:PRINT@468,"PRINTER NOT READY
    ":GOSUB1700:CF=2:RETURN
1670 IFPEEK(16425)<54THENRETURNELSECF=2:LPRINTCHR$(12):RETURN
1680 PRINT:PRINT"TYPE @ TO EXIT --> ";
1690 FL=-1:GOSUB140:IFCF=1THENRETURNELSEPRINTCHR$(8);:GOTO1690
1700 FORTT=1TO2500:NEXT:RETURN
1710 OPEN"I",1,PS$
1720 LINEINPUT#1,N$:LINEINPUT#1,A$:LINEINPUT#1,CZ$
1730 LINEINPUT#1,PN$:INPUT#1,FC!,GT,PD,PC,PT,PP:LINEINPUT#1,GL$
1740 CLOSE1:DIMG$(GT)
1750 OPEN"I",1,PI$
1760 INPUT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7,CV#
1770 FORN=1TOTN-1:INPUT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1
1780 OPEN"R",2,PD$:RETURN
1790 OPEN"O",1,PI$
1800 PRINT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7,STR$(CV#)+"D0"
1810 FORN=1TOTN-1:PRINT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1:RETURN
1820 IFERR/2+1=54THENCLS:PRINT@465,"NO FILES ON DATA DISK":PRINT:PRINT:CLEAR50:END
1830 IFERR/2+1=57THEN1860
1840 IFERR/2+1=70THENCLS:PRINT@465,"*** INVALID PASSWORD ***":PRINT:PRINTTAB(20)
    "PROGRAM TERMINATED":PRINT:CLEAR50:END
1850 PRINT:PRINTCHR$(30);"UNEXPECTED ERROR CODE ";ERR/2+1;"IN LINE";ERL:GOSUB168
    0:GOTO130
1860 TE=TE+1:IFTE>5THENCLS:PRINT@404,"** DISK I/O ERROR **":GOSUB1680:GOTO130ELSE
    ERESUME
1870 DATA ##,####,#####.##+,#####,#####.####.##,NOTHING DUE / CREDIT,CURRENT
    BALANCE DUE,30-60 DAYS OVERDUE,60-90 DAYS OVERDUE,PAST 90 DAYS OVERDUE,CRE
    DIT SUSPENDED,LEGAL ACTION PENDING,INACTIVE ACCOUNT,70476883289673379786192
    5
1880 DATA33,0,60,6,64,78,62,127,145,48,4,14,32,24,7,62,32,145,56,2,14,32,121,205
    ,59,0,35,16,232,62,13,205,59,0,124,254,64,32,220,201
1890 LPRINTLL$:LPRINTLF$:LPRINTLF$
1900 VS!=(PEEK(VARPTR(VS$)+1))+256*(PEEK(VARPTR(VS$)+2))
1910 IFVS!>32767THENVS!=VS!-65536
1920 DEFUSR0=VS!:VS=USR(0):LPRINTLF$:LPRINTLL$:LPRINTCHR$(12):RETURN

```

```
10 REM *** ARS END OF PERIOD PROCESSING ***
20 REM COPYRIGHT (C) 1979, TANDY CORP.
30 REM VERSION 1.0 - 07/16/79
40 CLEAR600:DEFINT A-Z:DIMP(500,2),W,IN$,CF,WD,W$,WS,WL,FL:W$="":N#=0:W#=0:V$="":
   X=0:D1#=16777216:D2#=65536:D3#=256
50 ONERRORGOTO2190:OPEN"I",1,"TRANSFER"
60 LINEINPUT#1,P1$:LINEINPUT#1,PD$
70 LINEINPUT#1,PS$:LINEINPUT#1,PT$:LINEINPUT#1,PG$
80 CLOSE1
90 POKE16424,67:POKE16425,1
100 READL0$,L1$,L2$,L3$,TG$,TH$,TI$,TJ$,TK$,TL$,TM$,TN$,OS$:RESTORE
110 XX=1:LE$="
   ":LF$=" ":LL$=STRING$(80,"-")
120 GOSUB2070:J=1:GOSUB1350:GOSUB2490:GOSUB440
130 IFEP=5THENEP=0
140 CLS:PRINT@469,"RETURNING TO MAIN MENU":GOSUB2160:CLOSE:CLEAR50:RUN"ARS":END
150 W=FRE(W$):IN$="":CF=0:W$=INKEY$:WD=0:WS=WD:WL=WD:IFFL=WDTHENFL=1
160 PRINTSTRING$(ABS(FL),CHR$(136)):STRING$(ABS(FL),CHR$(24)):
170 PRINTCHR$(14):FORW=1TO25:W$=INKEY$:IFW$<>" "THEN180ELSENEXT:PRINTCHR$(15):IF
   ORW=1TO25:W$=INKEY$:IFW$<>" "THEN180ELSENEXT:GOTO170
180 IFW$<>CHR$(13)THEN200ELSEPRINTSTRING$(ABS(FL)-WL," "):
190 PRINTCHR$(15):W=25:NEXT:RETURN
200 IFW$<>"@"THEN220
210 CF=1:PRINTCHR$(15):RETURN
220 PRINTCHR$(14):IFW$=CHR$(24)THENPRINTSTRING$(WL,CHR$(24)):GOTO150
230 IFW$<>CHR$(8)THEN270ELSEIFWL=0THEN170ELSEPRINTCHR$(24):IFFL>0THEN250ELSEIFP
   EEK(16418)=44THEN260
240 IFPEEK(16418)=46THENWD=0:GOTO250ELSEIFPEEK(16418)=43ORPEEK(16418)=45THENWS=0
250 IN$=LEFT$(IN$,LEN(IN$)-1)
260 WL=WL-1:POKE16418,136:GOTO170
270 IFABS(FL)=WLTHEN170ELSEIFFL>0THENIFW$>=" "ANDW$<="z"THEN320
280 IFW$="."ANDWD=0THENWD=1:GOTO320
290 IFW$=","THENPRINTW$:WL=WL+1:GOTO330
300 IF(W$="-"ORW$="+")ANDWS=0ANDWL=0THENWS=1:GOTO320
310 IFW$<"@"ORW$>"9"THEN170
320 PRINTW$:IN$=IN$+W$:WL=WL+1
330 IFABS(FL)=1THEN190ELSE170
340 IFIN$=" "THENDX!=0:RETURN
350 IFLEN(IN$)<>8ORMID$(IN$,3,1)<>"/"ORMID$(IN$,6,1)<>"/"THEN390
360 DX$=MID$(IN$,7,2)+MID$(IN$,1,2)+MID$(IN$,4,2)
370 FORW=1TO6:IFMID$(DX$,W,1)<"@"ORMID$(DX$,W,1)>"9"THENW=6:NEXT:GOTO390ELSENEXT
380 DX$=VAL(DX$):RETURN
390 FORW9=1TO4:PRINT@960,CHR$(30):FORW=1TO155:PRINT@975,"      INVALID FORMAT !";
   :NEXTW:FORW=1TO15:PRINT@975,CHR$(30):NEXTW:NEXTW9:CF=2:RETURN
400 DX$=STR$(DX!):DX$=RIGHT$(DX$,LEN(DX$)-1)
410 IFLEN(DX$)<6THENFORW=1TO6-LEN(DX$):DX$="@"+DX$:NEXT
420 IN$=MID$(DX$,3,2)+"/"+MID$(DX$,5,2)+"/"+MID$(DX$,1,2)
430 RETURN
440 CLS:PRINTTAB(15)"END OF PERIOD PROCESSING":PRINT
450 PRINT"OPTIONS:"
```

```

ERR(P)PRINT ALL STATEMENTS
ERR(U)PDATE ALL ACCOUNT RECORDS
ERR(T)RANSACTION RECORD PRINTOUT"
460 PRINT"ERR(C)LEAR SYSTEM FOR NEW MONTH
ERR(S)INGLE STATEMENT PRINTOUT
ERR(A)LIGN STATEMENTS ON PRINTER
ERR(D)EXIT TO MAIN MENU"
470 PRINT:GOSUB1870
480 PRINT@768,
    "NOTE: THE LAST THREE OPTIONS MAY BE USED AT ANY TIME. THE FIRST      FOUR
    OPTIONS MUST BE SELECTED IN SEQUENCE. THE ARROW ==>      POINTS TO THE
    NEXT REQUIRED STEP."
490 PRINT@960,"SELECT OPTION --> ";
500 FL=1:GOSUB150:IFCF<>0THENRETURNELSE0=INSTR("PVUTCSA",IN$)
510 IFO=0THEN490
520 ONOGOTO530,680,710,920,950,1000,2250
530 IFE>2THEN440ELSEPOKE16424,52:POKE16425,1:GOSUB2020:IFCF=2THEN440ELSEOPEN"R"
    ,3,PT$
540 CLS:PRINT"PRINT BY:

    (A)CCOUNT NUMBER

    (N)AME

    (Z)IP CODE

    ENTER SELECTION --> ";
550 FL=1:GOSUB150:IFCF<>0THENCLOSE3:GOTO440ELSEIFIN$<>"A"ANDIN$<>"N"ANDIN$<>"Z"TH
    ENPRINTCHR$(8):GOTO550
560 FA#:=0D0:IFIN$="A"THENAM=0 ELSEIFIN$="N"THENAM=1 ELSEAM=2
570 GOSUB2570
580 FORN=1TOTN-1:J=P(N,AM):IFSGN(J)=-1THEN670ELSEGOSUB1350:GOSUB1430
590 PRINT@576,"ACCOUNT : ";CVI(DD$):IF(CVD(DB$)=0ANDASC(DH$)=0)ORASC(DF$)=7THEN
    670ELSEGOSUB1440:AB#:=PB#:GOSUB1660
600 ONST+1GOSUB2640,2650,2660,2670,2680,2690,2700,2710
610 GOSUB1500
620 IFTC<>0THENGOSUB2580
630 AB#:=AB#+FC#:IFFC#>0THENGOSUB1630
640 IFSGN(CB#)=1THENGOSUB1610ELSEGOSUB1620
650 GOSUB1600
660 IFMR#:=0THENLPRINTTAB(50)"*****":LPRINTTAB(5)MS$:LPRINTLE$:ELSEIFMR#>0TH
    ENLPRINTTAB(50)USINGL2$:MR#:=LPRINTTAB(5)MS$:LPRINTLE$:ELSELPRINTTAB(50)"**C
    REDIT**":LPRINTTAB(5)MS$:LPRINTLE$
665 POKE16425,1
670 NEXT:CLOSE3:POKE16424,67:POKE16425,1:EP=1
680 IFEF<>1THEN440ELSECLS:PRINT@448,"DO YOU APPROVE OF THE THE STATEMENTS AS PRI
    NTED ? (Y/N) ";
690 FL=1:GOSUB150:IFCF<>0THEN440ELSEIFIN$<>"Y"ANDIN$<>"N"THEN680
700 IFIN$="Y"THENEP=2:GOTO440ELSEEP=0:GOTO440

```

```
710 IFEP<>2THEN440ELSEAM=0
720 CLS:PRINT@466,"UPDATING ALL ACCOUNT RECORDS":FA#=0D0
730 FORN=1TOTN-1:J=P(N,AM):IFSGN(J)=-1THEN760ELSEGOSUB1350:GOSUB1430
740 IF(CVD(DB$)=0ANDASC(DH$)=0)ORASC(DF$)=7THEN760ELSEGOSUB1440
750 GOSUB1660
760 NEXT:GOSUB2720:EP=3:GOTO440
770 IN$=STR$(VE#):GOSUB2390:VE$=V$:IN$=STR$(VF#):GOSUB2390:VF$=V$
780 IN$=STR$(VG#):GOSUB2390:VG$=V$:RETURN
790 OPEN"R",3,PT$:RETURN
800 GOSUB790:GOSUB2020:IFCF<>0THENCLOSE3:GOTO440
810 LPRINTLEFT$(TIME$,8);": COMPLETE TRANSACTION FILE (FINAL) - REPORT#":R6:LPRI
    NTLF$
820 GOSUB1300:L$=""
830 FORK=1TOUN-1:GOSUB1880:GOSUB1920:GOSUB1930
840 GOSUB880
850 GOSUB2030:IFCF=2THENCF=0:GOSUB1300
860 GOSUB1330
870 NEXT:CLOSE3:LPRINTLF$:LPRINT"END OF LISTING":LPRINTCHR$(12):EP=4:R6=R6+1:IFR
    6>32767THENR6=1:GOTO1130ELSE1130
880 IFVA$="S"THENVA$="SALE":A1=1 ELSEIFVA$="P"THENVA$="PMNT":A1=2 ELSEIFVA$="F"TH
    HENVA$="F/C":A1=3:VB=0:VD!=0:RM$="GL FILE":GOTO900ELSEIFVA$="B"THENVA$="B/D
    ":A1=2 ELSEVA$="B/D":A1=2:RM$="GL FILE":GOTO900
890 IFVI$="N"THENRM$="NORMAL"ELSEIFVI$="C"THENRM$="CORRECTION"ELSEIFVI$="A"THENR
    M$="SUPERSEDED"ELSEIFVI$="B"THENRM$="CORRECTION PENDING"
900 IFSGN(VB)=1THENRM$=RM$+"-UNPOSTED"ELSERM$=RM$+"-POSTED"
910 RETURN
920 IFEP<>3ANDEP<>4THEN440
930 CLS:PRINT@464,"PRINTING FINAL TRANSACTION FILE"
940 GOTO800
950 IFEP<>4THEN440ELSECLS:PRINT@465,"CLEARING SYSTEM FOR NEW MONTH":KILLPT$
960 OPEN"R",1,PG$
970 FORG=1TOGT:GOSUB1950:GOSUB1990:GOSUB2000:GC#=0:GOSUB1970
980 NEXT:CLOSE1:FORN=1TOTN-1:J=P(N,0):IFSGN(J)=-1THEN990ELSEGOSUB1350:GOSUB1430:
    LSETDI$=MKI$(0):LSETDJ$=MKI$(0):GOSUB1420
990 NEXT:EP=5:U=0:UT=0:UN=1:GOTO440
1000 POKE16424,52:POKE16425,1:GOSUB2020:IFCF=2THEN440ELSEOPEN"R",3,PT$:AM=0
1010 CLS:PRINT@460,"ENTER ACCOUNT #:"
1020 FL=-4:GOSUB150:IFCF<>0THENCLOSE3:GOTO440ELSEL$=IN$
1030 GOSUB1760:IFKS=0THENGOSUB2010:PRINTTAB(22)"NO SUCH ACCOUNT":GOSUB2060:CLOSE
    3:GOTO440
1035 IF(CVD(DB$)=0ANDASC(DH$)=0)ORASC(DF$)=7THEN1120
1040 AB#=PB#:GOSUB1660
1050 ONST+1GOSUB2640,2650,2660,2670,2680,2690,2700,2710
1060 GOSUB1500
1070 IFTC<>0THENGOSUB2580
1080 AB#=AB#+FC#:IFFC#>0THENGOSUB1630
1090 IFSGN(CB#)=1THENGOSUB1610ELSEGOSUB1620
1100 GOSUB1600
1110 IFMR#>0THENLPRINTTAB(50)USINGL2$:MR#:LPRINTTAB(5)MS$:LPRINTLE$ELSEIFMR#=0TH
    ENLPRINTTAB(50)"*****":LPRINTTAB(5)MS$:LPRINTLE$ELSELPRINTTAB(50)"**CR
```

```
=====
EDIT**":LPRINTTAB(5)MS$:LPRINTLE$
1120 CLOSE3:POKE16424,67:POKE16425,1:GOTO440
1130 OPEN"R",1,PG$:GOSUB2020:IFCF<>0THEN440
1140 LPRINTLEFT$(TIME$,8):" - GENERAL LEDGER FINAL RECAP ";
1150 LPRINT"(TOTALS THIS MONTH TO DATE) - REPORT #":R7
1160 LPRINTLL$:LPRINTLF$
1170 LPRINTTAB(20)"DEBIT(+)":TAB(50)"CREDIT(-)"
1180 LPRINTTAB(20)"-----":TAB(50)"-----"
1190 G=1:GOSUB1950:GOSUB1990:AR=CVI(G1$)
1210 LPRINTLF$:FORG=3TOGT:GOSUB1950:GOSUB1990:GOSUB2000
1220 AN$=MID$(GL$(G-1)*5+1,5)
1230 LPRINT"GLC:":G$:IFSGN(GC#)=-1THEN1260
1240 LPRINTTAB(10)AR:"/ A/R":TAB(25)USINGL2$:GC#:
1250 LPRINTTAB(40)GA:"/ ":AN$:TAB(55)USINGL2$:-GC#:GOTO1280
1260 LPRINTTAB(10)GA:"/ ":AN$:TAB(25)USINGL2$:-GC#:
1270 LPRINTTAB(40)AR:"/ A/R":TAB(55)USINGL2$:GC#
1280 NEXT:CLOSE1:R7=R7+1:IFR7>32767THENR7=1
1290 LPRINTLF$:LPRINT"END OF REPORT":LPRINTLL$:LPRINTCHR$(12):GOTO440
1300 LPRINTSTRING$(80,61)
1310 LPRINT"TYPE":TAB(9)"ACCT#":TAB(32)"DATE":TAB(43)"INV#":TAB(60)"REMARKS"
1320 LPRINTTAB(8)"AMOUNT (GLC)":TAB(26)"AMOUNT (GLC)":TAB(44)"AMOUNT (GLC)":TAB(
63)"TOTAL (GLC)":LPRINTSTRING$(80,61):RETURN
1330 LPRINTVA$:TAB(9)ABS(VB):TAB(32)VC$:TAB(42)VD!:TAB(60)RM$
1340 LPRINTTAB(4)USINGL2$:VE$:LPRINT(";VJ:");TAB(22)USINGL2$:VF$:LPRINT(";V
K:");TAB(40)USINGL2$:VG$:LPRINT(";VL:");TAB(58)USINGL2$:VH$:LPRINT(";
A1:");LPRINTLL$:RETURN
1350 JR=INT((J-1)/2)+1:JD=J-2*INT((J-1)/2)-1
1360 FIELD2,JD*127ASD$,11ASD0$,13ASD1$,17ASD2$,15ASD3$,5ASD4$,12ASD5$,4ASD6$,4AS
D7$,8ASD8$,8ASD9$,8ASDA$,8ASDB$,4ASDC$,2ASDD$,1ASDE$,1ASDF$,1ASDG$,1ASDH$,2
ASDI$,2ASDJ$:RETURN
1370 RQ$=CHR$(RQ):ST$=CHR$(ST):TC$=CHR$(TC):LSETD0$=F$:LSETD1$=L$:LSETD2$=SA$:LS
ETD3$=CS$:LSETD4$=Z$
1380 LSETD5$=PH$:LSETD6$=MKS$(DS!):LSETD7$=MKS$(DP!):LSETD8$=MKD$(PB#)
1390 LSETD9$=MKD$(TS#):LSETDA$=MKD$(PR#):LSETDB$=MKD$(CB#)
1400 LSETDC$=MKS$(CL!):LSETDD$=MKI$(ID):LSETDE$=RQ$
1410 LSETDF$=ST$:LSETDG$=FC$:LSETDH$=TC$:LSETDI$=MKI$(DI):LSETDJ$=MKI$(DJ)
1420 PUT2,JR:RETURN
1430 GET2,JR:RETURN
1440 F$=D0$:L$=D1$:SA$=D2$:CS$=D3$:Z$=D4$:PH$=D5$:DS!=CVS(D6$)
1450 DP!=CVS(D7$):PB#=CVD(D8$):TS#=CVD(D9$):PR#=CVD(DA$)
1460 CB#=CVD(DB$):CL!=CVS(DC$):ID=CVI(DD$):RQ$=DE$
1470 ST$=DF$:FC$=DG$:TC$=DH$:DI=CVI(DI$):DJ=CVI(DJ$)
1480 RQ=ASC(RQ$):ST=ASC(ST$):TC=ASC(TC$)
1490 RETURN
1500 IFPP=1THENLPRINTTAB(21)N$:LPRINTTAB(21)A$:LPRINTTAB(21)CZ$:LPRINTTAB(21)PN$
:LF=2:GOSUB1590ELSELF=6:GOSUB1590
1510 LPRINTTAB(45)USINGL0$:ID$:LPRINTTAB(54)EF$
1520 IFRIGHT$(F$,1)<>CHR$(0)THEN1540ELSEIFASC(F$)=0THENF$=CHR$(32):GOTO1540
1530 F$=LEFT$(F$,INSTR(F$,CHR$(0))-1)
1540 LPRINT" ":LPRINT" ":IFLEFT$(F$,1)=" "THENLPRINTTAB(15)L$:TAB(54)ED$:GOTO156
```

0

```

1550 LPRINTTAB(15)F$;" ";L$;TAB(54)ED$
1560 LPRINTTAB(15)SA$;LPRINTTAB(15)CS$;TAB(31)Z$
1570 LF=4:GOSUB1590:LPRINTTAB(5)"STATUS: ";SV$
1580 LF=21-PEEK(16425):GOSUB1590:LPRINTTAB(50)USINGL2$;AB#:RETURN
1590 FORLZ=1TOLF:LPRINT" ":NEXT:RETURN
1600 LF=48-PEEK(16425):GOSUB1590:LPRINTTAB(5)MF$;:RETURN
1610 LPRINTTAB(5)"TOTAL - REMIT THIS AMOUNT TO PAY IN FULL->";TAB(50)USINGL2$;CB
    #:RETURN
1620 LPRINTTAB(5)"TOTAL- YOU HAVE CREDIT IN THIS AMOUNT --->";TAB(50)USINGL2$;CB
    #:RETURN
1630 LPRINTTAB(15)"FIN. CHG.";TAB(24)USINGL2$;FC#;:LPRINTTAB(50)USINGL2$;AB#:RETU
    RN
1640 LPRINTTAB(5)VC$;TAB(16)VD!;TAB(24)USINGL2$;VH#;:LPRINTTAB(50)USINGL2$;AB#:R
    ETURN
1650 LPRINTTAB(5)VC$;TAB(15)"PAYMENT";TAB(36)USINGL2$;VH#;:LPRINTTAB(50)USINGL2$
    ;AB#:RETURN
1660 RQ#=INT(CDBL(RQ)*1000D0+.5D0)/1000D0:FI#=INT(CDBL(FC!)*1000D0+.5D0)/1000D
    0:FC#=0:IFSGN(PB#+PR#)=1ANDFC#="Y" THENFC#=INT((PB#+PR#)*FI#*100D0)/100D0:FA
    #=FA#+FC#
1670 CB#=PB#+TS#+PR#+FC#:IFPB#<MB#THENMP#=PB#ELSEMP#=PB#*RQ#/100D0
1675 IFMP#<MB#ANDPB#>=MB#THENMP#=MB#
1680 IFCB#<MB#THENMR#=CB#ELSEMR#=CB#*RQ#/100
1685 IFMR#<MB#ANDCB#>=MB#THENMR#=MB#
1690 IFST<4ANDABS(PR#)>MP#THENST=1:GOTO1730
1695 IFST<4ANDABS(ABS(PR#)-MP#)<=.009THENST=1:GOTO1730
1700 IFST<3ANDABS(PR#)<MP#THENST=ST+1:GOTO1730
1710 IFST=3THENST=4:MR#=CB#:GOTO1730
1720 MR#=CB#
1730 TS#=0:PR#=0:PB#=CB#:IFCB#<=0ANDST<5THENST=0
1740 IF0=3THENTC=0:GOSUB1370
1750 RETURN
1760 NL=0:NH=TN
1770 KS=INT((NH-NL)/2):IFKS=0THENRETURNELSEN=KS+NL
1780 IFN=NHTHENNH=INT((NH-NL)/2)+NL:GOTO1770ELSEIFP(N,AM)<0THENN=N+1:GOTO1780
1790 J=ABS(P(N,AM)):GOSUB1350:GOSUB1430:DD=CVI(DD$)
1800 IFAM=1THEN1830
1810 IFDD=VAL(L$)THENL$=DD$:GOTO1860
1820 IFDD>VAL(L$)THENNH=N:GOTO1770ELSE1850
1830 IFLEFT$(D1$,LEN(L$))=L$THENL$=D1$:GOTO1860
1840 IFLEFT$(D1$,LEN(L$))>L$THENNH=N:GOTO1770
1850 NL=N:GOTO1770
1860 IFSGN(P(N,AM))=-1THENKS=0:RETURNELSEGOSUB1440:RETURN
1870 PRINT@VAL(MID$(OS$,EP*3+1,3)), "==>";:RETURN
1880 KR=INT((K-1)/8)+1:KD=K-8*INT((K-1)/8)-1
1890 FIELD3,KD*31ASVV$,1ASV0$,2ASV1$,4ASV2$,4ASV3$,4ASV4$,4ASV5$,4ASV6$,1ASV7$,2
    ASV8$,2ASV9$,1ASVX$,1ASVY$,1ASVZ$:RETURN
1900 LSETV0$=VA$:LSETV1$=MKI$(VB):LSETV2$=MKS$(VC!):LSETV3$=MKS$(VD!):LSETV4$=VE
    $:LSETV5$=VF$:LSETV6$=VG$:LSETV7$=VI$:LSETVX$=VJ$:LSETVY$=VK$:LSETVZ$=VL$
1910 PUT3,KR:RETURN

```

```
-----
1920 GET3,KR:RETURN
1930 VA$=V0$:VB=CVI(V1$):VC!=CVS(V2$):VD!=CVS(V3$):VE$=V4$:VF$=V5$:VG$=V6$:VI$=V
7$:VJ=ASC(VX$):VK=ASC(VY$):VL=ASC(VZ$):VJ$=VX$:VK$=VY$:VL$=VZ$
1940 DX!=VC!:GOSUB400:VC$=IN$:V$=VE$:GOSUB2450:VE$=N$:V$=VF$:GOSUB2450:VF$=N$:V$
=VG$:GOSUB2450:VG$=N$:VH$=VE$+VF$+VG$:RETURN
1950 GR=INT((G-1)/25)+1:GD=G-25*INT((G-1)/25)-1
1960 FIELD1,GD*10ASG0$,2ASG1$,8ASG3$:RETURN
1970 LSETG1$=MKI$(GA):LSETG3$=MKD$(GC#)
1980 PUT1,GR:RETURN
1990 GET1,GR:RETURN
2000 GA=CVI(G1$):GC#=CVD(G3$):RETURN
2010 PRINTCHR$(29):CHR$(30):RETURN
2020 IFPEEK(14312)<128THENPOKE16425,1:RETURNELSECLS:PRINT@471,"PRINTER NOT READY
":GOSUB2060:CF=2:RETURN
2030 IFPEEK(16425)<54THENRETURNELSECF=2:LPRINTCHR$(12):RETURN
2040 PRINT:PRINT"TYPE @ TO EXIT --> ";
2050 FL=-1:GOSUB150:IFCF=1THENRETURNELSEPRINTCHR$(8):GOTO2050
2060 FORTT=1TO2500:NEXT:RETURN
2070 OPEN"I",1,PS$
2080 LINEINPUT#1,N$:LINEINPUT#1,A$:LINEINPUT#1,CZ$
2090 LINEINPUT#1,PN$:INPUT#1,FC!,GT,PD,PC,PT,PP:LINEINPUT#1,GL$
2100 INPUT#1,MB#
2110 CLOSE1:DIMG$(GT)
2120 OPEN"I",1,PI$
2130 INPUT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7,CV#
2140 FORN=1TOTN-1:INPUT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1
2150 OPEN"R",2,PD$:RETURN
2160 OPEN"O",1,PI$
2170 PRINT#1,TI,TN,F,U,UT,UN,EP,R0,R1,R2,R3,R4,R5,R6,R7:STR$(CV#)+"D0"
2180 FORN=1TOTN-1:PRINT#1,P(N,0),P(N,1),P(N,2):NEXT:CLOSE1:RETURN
2190 IFERR/2+1=54THENCLS:PRINT@468,"NO ACCOUNTS ON DATA DISK":PRINT:PRINT:CLEAR5
0:END
2200 IFERR/2+1=57THEN2230
2210 IFERR/2+1=70THENCLS:PRINT@465,"*** INVALID PASSWORD ***":PRINT:PRINTTAB(20)
"PROGRAM TERMINATED":PRINT:CLEAR50:END
2220 PRINT:PRINTCHR$(30):"UNEXPECTED ERROR CODE ";ERR/2+1;" IN LINE":ERL:GOSUB204
0:GOTO140
2230 TE=TE+1:IFTE>5THENCLS:PRINT@404,"** DISK I/O ERROR **":GOSUB2040:GOTO140ELS
ERESUME
2240 DATA"#####","#####","#####.##+","###.#"
2250 GOSUB2020:IFCF=2THEN440ELSEPOKE16424,52:POKE16425,1:CLS
2260 PRINT@448,"
(H)ORIZONTAL ALIGNMENT TEST
(V)ERTICAL ALIGNMENT TEST
(A)LIGNMENT TEST FINISHED

ENTER SELECTION --> ";
2270 AL$=INKEY$:IFAL$<>"H"ANDAL$<>"V"ANDAL$<>"A"THEN2270
2280 IFAL$="A"THENPOKE16424,67:POKE16425,1:GOTO440ELSEIFAL$="H"THENLPRINT" -><-
CENTER THE X'S ON THE PERFORATIONS -><-":GOTO2260
```

```
2290 LPRINTCHR$(12):GOTO2260
2300 DATA"NOTHING DUE / CREDIT"
2310 DATA"CURRENT BALANCE DUE"
2420 DATA"30-60 DAYS OVERDUE"
2330 DATA"60-90 DAYS OVERDUE"
2340 DATA"PAST 90 DAYS OVERDUE"
2350 DATA"CREDIT SUSPENDED"
2360 DATA"LEGAL ACTION PENDING"
2370 DATA"INACTIVE ACCOUNT"
2380 DATA"256000320384448640"
2390 N#=VAL(IN$):W#=ABS(N#)*100:V$=""
2400 X=W#/D1#:W#=W#-X*D1#:V$=V$+CHR$(X-(N#<0)*128)
2410 X=W#/D2#:W#=W#-X*D2#:V$=V$+CHR$(X)
2420 X=W#/D3#:W#=W#-X*D3#:V$=V$+CHR$(X)
2430 V$=V$+CHR$(W#)
2440 RETURN
2450 W#=(ASC(MID$(V$,1,1))*D1#+ASC(MID$(V$,2,1))*D2#+ASC(MID$(V$,3,1))*D3
    #+ASC(MID$(V$,4,1)))
2460 IFASC(LEFT$(V$,1))>127THENW#=-W#
2470 N#=W#/100
2480 RETURN
2490 CLS:PRINT@982,"(PRESS <@> TO REDO)":PRINT@384,"ENTER THIS MONTH'S MESSAGE
    (2 LINES): ";
2500 MF$=" ":MS$=" ":PRINT@448,"1ST LINE: ";
2510 FL=41:GOSUB150:IFCF<>0THEN2490ELSEIFIN$<>" "THENMF$=IN$
2520 PRINT@512,"2ND LINE: ";
2530 FL=41:GOSUB150:IFCF<>0THEN2490ELSEIFIN$<>" "THENMS$=IN$
2540 PRINT@640,"PERIOD ENDING DATE (PRESS <ENTER> FOR TODAY'S DATE):"
2550 FL=8:GOSUB150:IFCF<>0THEN2490ELSEIFIN$=" "THENPRINTCHR$(29):ED$=LEFT$(TIME$
    ,8):PRINTED$ELSEED$=IN$
2560 GOSUB340:IFCF=2THEN2540
2562 PRINT@768,"BILLING DATE (PRESS <ENTER> FOR TODAY'S DATE):"
2564 FL=8:GOSUB150:IFCF<>0THEN2490ELSEIFIN$=" "THENPRINTCHR$(29):EF$=LEFT$(TIME$
    ,8):PRINTEF$ELSEEF$=IN$
2566 GOSUB340:IFCF=2THEN2562ELSERETURN
2570 CLS:PRINT@472,"PRINTING STATEMENTS":RETURN
2580 K=DI
2590 GOSUB1880:GOSUB1920:Q=CVI(V9$):IFV7$="A"ORV0$="B"THEN2620ELSEGOSUB1930
2600 AB#=AB#+VH#:IFVA$="S"THENGOSUB1640ELSEGOSUB1650
2610 IFPEEK(16425)>46THENGOSUB2790
2620 IFQ=0THEN2630ELSEK=Q:GOTO2590
2630 IFPEEK(16425)>45THENGOSUB2790:RETURNELSERETURN
2640 SV$=TG$:RETURN
2650 SV$=TH$:RETURN
2660 SV$=TI$:RETURN
2670 SV$=TJ$:RETURN
2680 SV$=TK$:RETURN
2690 SV$=TL$:RETURN
2700 SV$=TM$:RETURN
2710 SV$=TN$:RETURN
```

```
2720 OPEN"R",3,PT$
2730 IN$=LEFT$(TIME$,8):GOSUB340:VC!=DX!:VD!=-1:VA$="F"
2740 VB=-1:VE#=FA#:VF#=0:VG#=0:VH#=FA#:VJ$=CHR$(3):VK$=CHR$(0):VL$=CHR$(0):VI$="
N":GOSUB770:CV#=CV#+FA#
2750 K=UN:GOSUB1880:GOSUB1920:GOSUB1900:CLOSE3:UN=UN+1:UT=UT+1
2760 OPEN"R",1,PG$
2770 G=1:GOSUB1950:GOSUB1990:GOSUB2000:GC#=GC#+FA#:GOSUB1970
2780 G=3:GOSUB1950:GOSUB1990:GOSUB2000:GC#=GC#+FA#:GOSUB1970:CLOSE1:RETURN
2790 LF=48-PEEK(16425):GOSUB1590
2800 LPRINTTAB(10)"CONTINUED ON NEXT PAGE";
2810 LPRINTTAB(50)"*****"
2820 LPRINTL$:POKE16425,1:LF=7-PEEK(16425):GOSUB1590
2830 LPRINTTAB(45)USINGL0$:ID$:LPRINTTAB(54)EF$
2840 LF=14-PEEK(16425):GOSUB1590
2850 LPRINTTAB(20)"CONTINUED FROM PREVIOUS PAGE"
2860 GOSUB1580:RETURN
```



### IMPORTANT NOTICE

ALL RADIO SHACK COMPUTER PROGRAMS ARE DISTRIBUTED ON AN "AS IS" BASIS WITHOUT WARRANTY.

Radio Shack shall have no liability or responsibility to customer or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by computer equipment or programs sold by Radio Shack, including but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of such computer or computer programs.

NOTE: Good data processing procedure dictates that the user test the program, run and test sample sets of data, and run the system in parallel with the system previously in use for a period of time adequate to insure that results of operation of the computer or program are satisfactory.

RADIO SHACK  A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102  
CANADA: BARRIE, ONTARIO L4M 4W5

---

#### TANDY CORPORATION

##### AUSTRALIA

280-316 VICTORIA ROAD  
RYDALMERE, N.S.W. 2116

##### BELGIUM

PARC INDUSTRIEL DE NANINNE  
5140 NANINNE

##### U. K.

BILSTON ROAD WEDNESBURY  
WEST MIDLANDS WS10 7JN

PRINTED IN U.S.A.